Chapitre 5

Codes correcteurs linéaires

Nous introduisons rapidement les éléments de base de la théorie des codes correcteurs classiques. Nous nous intéressons seulement aux codes correcteurs linéaires sur un alphabet binaire dans ce cours. Nous introduisons les matrices génératrices et de parité ainsi que leurs propriétés importantes. Nous discutons rapidement des codes duaux ainsi que les coensembles de code. Nous décrivons ensuite un code correcteur classique simple, appelé *code de Hamming*, qui corrige une erreur dans un bloc de taille n. Nous montrons une borne simple sur le taux de transmission maximal des codes capables de corriger un taux d'erreur $0 \leq p \leq \frac{1}{2}$. Nous montrons ensuite que les codes linéaires aléatoires peuvent corriger un taux d'erreur p sur un canal symétrique binaire à un taux de transmission qui n'est pas très loin de la capacité du canal.

Nous montrons ensuite comment construire un code quantique simple : le code de Shor. Il permet de corriger une renversement de bit et un renversement de phase sur 9 qubits. Le code de Shor fait mieux, car tous les code correcteurs quantiques qui permettent de corriger t renversements de bit et t renversements de phase permettent également de corriger n'importe quelle opération quantique qui opère sur au plus t positions. Pour développer des codes correcteurs quantiques il suffit de trouver des façons de corriger pour les renversements de Pauli. Nous terminons avec quelques bornes supérieures sur le taux de transmission de codes quantiques capables de corriger un certain taux d'erreur.

5.1 Codage par bloc

Un *code par bloc* C sur alphabet Σ avec paramètre n est un sous-ensemble des chaînes de longueur n composées de symboles dans Σ. Les éléments du code $C \subseteq \Sigma^n$ sont appelés *mots de code*. Nous dénotons par $GF(2) := \{(\oplus, \cdot), \{0,1\}\}$ le corps de Galois [1] à deux éléments dont l'addition "$\oplus$" est la somme modulo 2 et la multiplication "$\cdot$" correspond au *"ET"* logique. Les éléments $0 \in GF(2)$ et $1 \in GF(2)$ sont les identités additive et multiplicative dans $GF(2)$ respectivement. Un code $C \subseteq \{0,1\}^n$ est appelé *code binaire*. Les codes que nous verrons seront

1. Par abus de notation, nous écrirons à l'occasion $a \in GF(2)$ pour $a \in \{0,1\}$.

toujours des codes binaires. Les mots de code d'un code binaire par bloc C avec paramètre n peuvent être représentés par des vecteurs [2] dans $GF(2)^n$. $GF(2)^n$ est un espace euclidien où pour $u = (u_1, u_2, \ldots, u_n), v = (v_1, v_2, \ldots, v_n) \in GF(2)^n$ l'addition, dénotée '$\oplus$', est définie par

$$u \oplus v := (u_1 \oplus v_1, u_2 \oplus v_2, \ldots, u_n \oplus v_n) \ .$$

La multiplication par un scalaire $\alpha \in GF(2)$ est quant à elle définie par $\alpha \cdot u := (\alpha \cdot u_1, \alpha \cdot u_2, \ldots, \alpha \cdot u_n)$. Le produit scalaire entre $u, v \in GF(2)^n$ est défini de façon naturelle comme

$$\langle u, v \rangle := \bigoplus_{i=1}^{n} u_i \cdot v_i \ .$$

Les propriétés et définitions que nous verrons peuvent facilement être adaptées au cas où les codes opèrent sur des corps finis de plus de deux éléments.

Définition 74. *Pour $u, v \in \{0,1\}^n$, la* distance de Hamming *entre u et v est définie par $\Delta_{\mathsf{H}}(u, v) := \#\{i \,|\, u_i \neq v_i\}$. Le* poids de Hamming *du vecteur u est défini par $\mathsf{w}(u) := \#\{i \,|\, u_i \neq 0\}$. La distance et le poids de Hamming peuvent être définis mutatis mutandis pour $u, v \in \Sigma^n$ où Σ contient les éléments d'un corps fini.*

Il est facile de vérifier que pour $u, v \in \{0,1\}^n$, nous avons :

$$\Delta_{\mathsf{H}}(u, v) = \mathsf{w}(u \oplus v) \ .$$

Cette égalité tient en général pour des chaînes d'éléments d'un corps fini arbitraire via $\Delta_{\mathsf{H}}(u, v) = \mathsf{w}(u - v)$.

Un code binaire linéaire $C \subseteq \{0,1\}^n$ est un sous-espace linéaire de $GF(2)^n$. Une propriété essentielle des codes correcteurs qui permet de caractériser sa capacité à corriger les erreurs est sa *distance minimum* définie de la façon suivante :

Définition 75. *Soit $C \subseteq \{0,1\}^n$ un code binaire, la* distance minimum *d de C est définie par*

$$d := \min_{c \neq c' \in C} \Delta_{\mathsf{H}}(c, c') \ .$$

Si le code C est un code linéaire, il est facile de vérifier que

$$d = \min_{c \in C \setminus \{0^n\}} \mathsf{w}(c) \ . \tag{5.1}$$

Exercice 209. *Démontrez l'équation (5.1).*

2. Tandis que nous avons représenté les vecteurs dans un espace d'Hilbert comme des vecteurs colonnes, nous représentons les vecteurs dans $GF(2)^n$ comme des vecteurs lignes. C'est la façon habituelle de procéder pour représenter des chaînes de bits en vecteurs.

Lorsque $C \subseteq \Sigma^n$ est un sous-espace vectoriel (défini par un corps de Galois avec les opérations d'addition et de multiplication sur les éléments de Σ), nous dirons alors que C est un *code linéaire*. Un code $C \subseteq \{0,1\}^n$ est linéaire si pour chaque paire $c, c' \in C$ nous avons que $c \oplus c' \in C$. Une conséquence est que tout code linéaire contient le mot nul 0^n, car pour $c \in C$ nous avons $c \oplus c = 0^n \in C$. La dimension du code C, dénotée $\dim(C)$, est sa dimension en tant qu'espace vectoriel. Le nombre de mots de code de C est $\#C$ et satisfait $\#C = 2^{\dim(C)}$. Les mêmes observations peuvent être appliquées mutatis mutandis aux codes linéaires sur un alphabet $GF(p^t)$ quelconque pour p premier et $t \in \mathbb{N}^*$ (les corps ont toujours un nombre d'éléments qui correspond à une puissance entière d'un nombre premier).

Un code correcteur binaire et linéaire est caractérisé par 3 paramètres. Le premier est la longueur des mots de code, le second est la dimension du code comme espace vectoriel et la troisième est sa distance minimum. Plus précisemment :

Définition 76. *Un code binaire linéaire par bloc $C \subseteq \{0,1\}^n$ avec $\dim(C) = k$ et distance minimum au moins d est dénoté $[n,k,d]$–*code*. Le paramètre n est la* taille de bloc *de C. Un code linéaire par bloc $C \subseteq \Sigma^n$ sur alphabet Σ avec $\#\Sigma > 2$ de dimension $\dim(C) = k$ et de distance minimum d est dénoté par (n,k,d)–*code*. Le ratio $\frac{k}{n}$ d'un $[n,k,d]$–code C (ou d'un (n,k,d)–code) est appelé* taux de transmission *de C.*

La dimension $\dim(C) = k$ du code $C \subseteq \{0,1\}^n$ est le nombre de bits de message qui peuvent être encodés par C. Un encodage est simplement une façon d'associer à chaque message $m \in \{0,1\}^k$ son mot de code $c(m) \in C$. Le taux de transmission $\frac{k}{n}$ du code $C \subseteq \{0,1\}^n$ est la quantité de bits d'information sur le message m qu'un bit physique du mot de code transporte.

Si la distance minimum de C n'est pas connue ou n'est pas utile dans le contexte alors nous dirons de C qu'il est un $[n,k]$–code linéaire. Un $[n,k]$–code linéaire C est un sous-espace vectoriel de $GF(2)^n$ de dimension k. Considérez maintenant l'application $\mathcal{G} : \{0,1\}^k \to \{0,1\}^n$ qui associe à un mot $m \in \{0,1\}^k$ un mot de code $\mathcal{G}(m) \in C$. Pour $m, m' \in \{0,1\}^m$, un code linéaire demande que

$$\mathcal{G}(m) \oplus \mathcal{G}(m') = c \oplus c' \in C \ .$$

$\mathcal{G}$ peut être vue comme une application linéaire en posant

1. $\mathcal{G}(0^k) = 0$, ce qui assure que pour chaque $\lambda \in GF(2)$ et $m \in GF(2)^k$, $\mathcal{G}(\lambda m) = \lambda \mathcal{G}(m)$ et
2. $\mathcal{G}(m \oplus m') = \mathcal{G}(m) \oplus \mathcal{G}(m')$ en conservant le même code C, car la façon dont nous associons les messages $m \in \{0,1\}^k$ aux mots du code C n'est pas importante. N'importe quel choix de la base d'expression des messages produira le même code correcteur d'erreur.

Au point 2, la base canonique $\{|e_i\rangle\}_{i=1}^k \in 2^{GF(2)^k}$ peut jouer le rôle de la base d'expression. Nous avons vu au théorème 1 que toute application linéaire $\mathcal{G} : \{0,1\}^k \to \{0,1\}^n$ peut être représentée par une matrice booléenne [3] $G \in \mathcal{L}_{k,n}(GF(2))$ telle que

$$\mathcal{G}(m) = m \cdot G \ .$$

3. Dans le but de simplifier la notation par la suite, nous utilisons ici une version équivalente du théorème 1 : à chaque application linéaire $\mathcal{G} : GF(2)^k \to GF(2)^n$ correspond une matrice $G \in \mathcal{L}_{k,n}(GF(2))$ telle que pour chaque $m \in \{0,1\}^k$, $\mathcal{G}(m) = mG$ où m est un vecteur rangée de k bits.

De cette façon, le code C est défini par l'action de la matrice G sur les messages de k bits :

$$C := \{x \cdot G | x \in \{0,1\}^k\} \ . \tag{5.2}$$

La matrice G est appelée *matrice génératrice* de C. Comme nous l'avons vu au théorème 1, un $[n,k]$–code C peut être décrit par plusieurs matrices génératrices.

Nous pouvons associer à la matrice génératrice $G \in \mathcal{L}_{k,n}(GF(2))$ d'un $[n,k]$–code C, une autre matrice booléenne $H \in \mathcal{L}_{n-k,n}(GF(2))$ qui satisfait la relation suivante :

$$G \cdot H^T = 0 \ . \tag{5.3}$$

Il s'agit de construire H avec des rangées orthogonales à celles de G. Ceci est toujours possible, car $\text{img}(G)$ est un sous-espace linéaire de dimension $\dim(\text{img}(G)) = k$ dans un espace linéaire de dimension n. Nous pouvons donc trouver $n-k$ vecteurs orthogonaux dans $GF(2)^n$ qui engendrent le sous-espace de dimension $n-k$ orthogonal à C. La matrice H est alors obtenue en plaçant ces $n-k$ vecteurs orthogonaux comme rangées de H.

Une matrice boléenne H qui satisfait (5.3) pour une matrice génératrice G associé à un code $C \subseteq \{0,1\}^n$ est appelée *matrice de parité* pour le code C. Le code C peut être défini par sa matrice de parité H en plus d'être défini par sa matrice génératrice G comme le montre l'équation (5.2) :

$$C = \{xG | x \in \{0,1\}^k\} = \{c \in \{0,1\}^n | cH^T = 0^{n-k}\} \ .$$

C'est donc dire qu'un code C est l'image de sa matrice génératrice et le noyau de sa matrice de parité.

La distance minimum d'un code C avec matrice de parité H est le plus grand entier d tel que n'importe quel ensemble de $d-1$ colonnes de H sont linéairement indépendantes mais un ensemble de d colonnes de H existe qui ne sont pas linéairement indépendantes.

Théorème 37. *Soit $H \in \mathcal{L}_{n-k,n}(GF(2))$ une matrice de parité pour un $[n,k,d]$–code. N'importe quel ensemble de $d-1$ colonnes de H sont linéairement indépendantes et il existe une ensemble de d colonnes de H qui sont linéairement dépendantes.*

Exercice 210. *Prouvez le théorème 37.*

En utilisant (5.3), nous obtenons que pour chaque $c \in C$ et $e \in \{0,1\}^n \setminus C$,

$$(c \oplus e)H^T = cH^T \oplus eH^T = 0^{n-k} \oplus eH^T = eH^T \neq 0^{n-k} \ . \tag{5.4}$$

Pour $c \in C \subseteq \{0,1\}^n$ un mot de code arbitraire et $e \in \{0,1\}$ un vecteur quelconque. Le mot $c \oplus e$ représente un mot de code dont les bits aux positions $1 \leq i \leq n$ telles que $e_i = 1$ sont renversés. Le vecteur e est le vecteur d'erreurs associé à la communication de $c \in C$ sur un canal bruité.

Définition 77. *Soit $C \subseteq \{0,1\}^n$ un code binaire avec matrice de parité $H \in \mathcal{L}_{n-k,n}(GF(2))$ et soit $\hat{c} \in \{0,1\}^n$. La chaîne $s := \hat{c}H^T \in \{0,1\}^{n-k}$ est appelée* syndrome de $\hat{c}$ selon H.

Exercice 211. *Soit $s \in \{0,1\}^{n-k}$ et $H \in \mathcal{L}_{n-k,n}(GF(2))$ une matrice de parité. Quels sont les mots $\hat{c} \in \{0,1\}^n$ de syndrome s exprimés de la façon la plus simple possible ?*

5.2 Correction et détection d'erreur

Les codes-correcteurs sont utilisés pour transmettre de l'information sur un canal bruité. Le receveur pourra récupérer intégralement l'information transmise par un mot de code bruité lorsque le nombre de bits erronés est moindre que $d/2$ où d est la distance minimum du code. La transmission de $c \in C$ sur un canal symétrique binaire complémente avec probabilité p chaque bit c_i en position $1 \leq i \leq n$ d'une façon indépendante. Les erreurs produites pendant la transmission peuvent être représentées par une chaîne $e \in \{0,1\}^n$ telle que le mot reçu $\hat{c} \in \{0,1\}^n$ satisfait

$$\hat{c} = c \oplus e \ .$$

Le syndrome $s = \hat{c}H^T = eH^T$ du mot reçu correspond donc au syndrome de la chaîne des erreurs $e \in \{0,1\}^n$ selon H. Notez que pour chaque paire de chaînes d'erreurs $e \neq e' \in \{0,1\}^n$ telle que $\mathsf{w}(e), \mathsf{w}(e') < \frac{d}{2}$, nous avons $eH^T \neq e'H^T$.

Exercice 212. *Montrez cet énoncé, pour $e \neq e' \in \{0,1\}^n$ tels que $\mathsf{w}(e) + \mathsf{w}(e') < \frac{d}{2}$ nous avons $eH^T \neq e'H^T$ où H est la matrice de parité d'un code linéaire C de distance minimum d.*

Notez que puisque $s = \hat{c}H^T = (c \oplus e)H^T = eH^T$, le syndrome s est la somme des rangées de H^T aux positions i tel que $e_i = 1$. Autrement dit, le syndrome correspond à la somme des colonnes de H aux positions où une erreur de transmission s'est produite.

L'exercice 212 nous permet maintenant de voir comment un $[n,k,d]$–code linéaire C peut être utilisé pour transmettre une message binaire de k bits qui pourra être récupéré intégralement par le receveur lorsque moins de $\frac{d}{2}$ renversements de bit sont produits pendant la transmission. Montrons comment y parvenir lorsque le code C est généré par la matrice génératrice $G \in \mathcal{L}_{k,n}(GF(2))$ et sa matrice de parité est $H \in \mathcal{L}_{n-k,n}(GF(2))$.

1. Alice veut transmettre le message $m \in \{0,1\}^k$ à Bob.
2. Alice calcule $c = mG \in \{0,1\}^n$ et transmet c sur le canal bruité.
3. Bob reçoit $\hat{c} \in \{0,1\}^n$.
4. Bob calcule le syndrome $s = \hat{c}H^T$ de $\hat{c}$ selon H.
5. Bob cherche le seul vecteur d'erreur $e^* \in \{0,1\}^n$ de poids $\mathsf{w}(e^*) < \frac{d}{2}$ tel que $e^*H^T = s$. Bob décode $\hat{c}$ en calculant $c' = \hat{c} \oplus e^*$.
6. Bob retrouve $m' \in \{0,1\}^k$ tel que $c' = m'G$.

La correction d'erreur effectuée par Bob ici est appelée *décodage au plus proche*. Elle consiste à trouver le mot de code $c' \in C$ le plus proche du mot reçu $\hat{c} \in \{0,1\}^n$. Cette manière de décoder produit le mot de code le plus probable lorsque, en particulier, le bruit de transmission renverse les bits du mot de code transmis indépendamment les uns des autres avec probabilité constante moindre que $\frac{1}{2}$. Le canal symétrique binaire avec paramètre $p < \frac{1}{2}$ est de ce type. La méthode de décodage de Bob est donc celle qui minimise la probabilité d'erreur lorsque la transmission se déroule sur un canal symétrique binaire.

5.2.1 Correction d'erreur

La correction d'erreur décrite à l'étape 5 plus haut, permet à Bob de corriger à l'aide de C n'importe quel $e \in \{0,1\}^n$ (i.e. le schéma des erreurs) avec $\mathrm{w}(e) < \frac{d}{2}$. Dans ce cas, Bob retrouve toujours $c' = c$ et donc $m' = m$. La recherche du vecteur d'erreur $e^* \in \{0,1\}^n$ de moindre poids de Hamming pour un syndrome d'erreur donné est une tâche lourde nécessaire pour réaliser le décodage au plus proche. Une façon de rendre le décodage plus performant est de ranger pour chaque syndrome $s \in \{0,1\}^{n-k}$, le vecteur d'erreur de moindre poids dans une table. La correction d'erreur devient alors une simple recherche dans la table du vecteur d'erreur associé au syndrome obtenu. La table peut être volumineuse, car elle correspond au nombre de syndromes possibles 2^{n-k}.

Le design de bons codes correcteurs d'erreur consiste justement à définir des codes avec bons taux de transmsmission pour lesquels la correction d'erreurs puisse se faire efficacement. Pour y parvenir, des codes avec certaines structures algébriques sont construits pour permettre le décodage au plus proche en tirant partie de la structure pour y parvenir efficacement. Cette façon de procéder se fait souvent au prix d'une réduction du taux de transmission. La théorie des codes correcteur à justement pour but le développement de codes qui permettent la correction d'erreur efficace au meilleur taux de transmission.

5.2.2 Détection d'erreur

Un $[n,k,d]$–code C peut être utilisé simplement pour détecter des erreurs de transmission sans nécessairement chercher à les corriger. Cette tâche est simple, elle ne demande que de vérifier si le syndrome reçu est nul ou pas, si la chaîne reçue $\hat{c}$ est telle que $\hat{c}H^T \neq 0$ alors au moins une erreur est survenue.

Si la distance minimum du code C et d, toute chaîne d'erreur $e \in \{0,1\}^n$ avec $0 < \mathrm{w}(e) < d$ produit nécessairement $\hat{c} = c \oplus e \notin C$. Le syndrome d'erreur résultant $s = eH^T$ ne pourra donc jamais être tel que $s = 0$ et la détection d'erreur de transmission sera faite avec succès.

5.2.3 Décodage final

Le décodage final qui permet à Bob de récupérer $m' \in \{0,1\}^k$ à partir de $c' \in C$ peut toujours être défini de telle façon que $m' = c'_1, c'_2, \ldots, c'_k$ en choisissant la matrice génératrice de la bonne façon. Soit $G \in \mathcal{L}_{k,n}(GF(2))$ une matrice génératrice arbitraire pour le code linéaire C. Puisque G contient des rangées linéairement indépendantes, nous pouvons appliquer l'élimination de Gauss-Jordan pour obtenir une matrice $G' = (g'_{ij})_{\substack{1\leq i\leq k\\ 1\leq j\leq n}}$ pour le même code dont la sous-matrice carrée satisfait $(g'_{ij})_{\substack{1\leq i\leq k\\ 1\leq j\leq k}} = \mathbb{1}_k$.

Définition 78. *Un $[n,k]$–code C avec une matrice génératrice $G = (g_{i,j})_{\substack{1\leq i\leq k\\ 1\leq j\leq n}}$ qui satisfait $(g_{ij})_{\substack{1\leq i\leq k\\ 1\leq j\leq k}} = \mathbb{1}_k$, $G = (\mathbb{1}_k | \tilde{G})$ pour $\tilde{G} \in \mathcal{L}_{k,n-k}(GF(2))$ est dit sous* forme systématique *ou* forme standard. *La*

matrice génératrice du code copie alors les k bits du message aux k premières positions du mot code correspondant.

Puisque n'importe quel code peut être mis sous forme systématique (voir exercice 213), il est toujours possible d'encoder l'information de sorte que les k premières positions du mot de code contiennent le message transmis sans nécessiter d'autres traitements. Lorsque G est sous forme systématique, une matrice de parité correspondante H est facile à dériver. En effet, si $G = (\mathbb{1}_k \,|\, \tilde{G})$ alors $H = (\tilde{G}^T \,|\, \mathbb{1}_{n-k})$. Nous avons bien alors, en accord avec (5.3), que

$$GH^T = (\mathbb{1}_k \,|\, \tilde{G}) \begin{pmatrix} \tilde{G} \\ \hline \mathbb{1}_{n-k} \end{pmatrix} = \tilde{G} \oplus \tilde{G} = 0 \ .$$

Lorsque H est de la forme $H = (\tilde{G}^T \,|\, \mathbb{1}_{n-k})$ nous disons que le matrice de parité est sous *forme systématique*.

Exercice 213. *Prouvez formellement que pour chaque $[n,k,d]$–code C il existe un $[n,k,d]$–code C' sous forme systématique.*

Il est noter que la mise d'un code sous forme systématique peut rendre le correction des erreurs moins efficace.

5.3 Code répétition

Voyons maintenant un code linéaire extrêmement simple et un peu naïf, le code 3–*répétition*. Il s'agit du code qui copie trois fois un message d'un seul bit. C'est un $[3,1,3]$–code, permettant donc de corriger une seule erreur de transmission arbitraire. Il est facile de vérifier que la matrice génératrice du code 3–répétition est

$$G = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \ .$$

La matrice de parité associée est donc

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \ .$$

Il s'agit d'un code linéaire dont le bit de message apparaît en première position (ainsi qu'en deuxième et en troisième) du mot de code correspondant, car la matrice génératrice est sous forme systématique. La matrice de parité est également sous forme systématique car elle contient la sous-matrice identité $(n-k) \times (n-k) = 2 \times 2$.

Exercice 214. *Supposons une transmission d'un mot du code 3–répétition sur un canal symétrique binaire avec paramètre $0 \leq p < \frac{1}{2}$. Donnez la valeur maximale de p pour que la transmission résulte en un bit décodé qui soit erroné avec probabilité moindre que p. Autrement dit, trouvez le paramètre maximal d'un canal symétrique binaire pour lequel le code 3–répétition améliore la fiabilité du bit transmis sur celui-ci.*

Le code 3-répétition peut être facilement généralisé en code n-répétition, produisant un $[n,1,n]$–code capable de corriger $\lfloor \frac{n-1}{2} \rfloor$ erreurs à tous coups.

Exercice 215. *Donnez la matrice génératrice et une matrice de parité pour le code n–répétition.*

Nous pouvons modifier la matrice génératrice du code 3–répétition pour que le syndrome donne directement la position de l'erreur lorsqu'il y en a une seule. Pour ce faire, considérez

$$H' = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} .$$

Exercice 216. *Montrez que H' est une matrice de parité pour le code 3–répétition.*

Il est facile de voir que le code 3–répétition produit un syndrome qui indique directement la position de l'erreur. Si le mot reçu est $c' = c \oplus e$ avec $\mathsf{w}(e) = 1$ et $e_i = 1$ alors $s = (s_1, s_2) = c'H'^T = eH'^T$ (où $s \in \{0,1\}^2$ et $s_1, s_2 \in \{0,1\}$) est tel que $i = \mathsf{decimal}(s)$ où $\mathsf{decimal}(s) = 2s_1 + s_2$ est la valeur décimale de la chaîne [4] binaire s. Ceci est facile à constater puisqu'une seule erreur retourne la colonne de H' correspondant à la position i de l'erreur. Or, la première colonne de H' contient la chaîne 10 (i.e. $\mathsf{decimal}(10) = 1$), la deuxième 01 (i.e. $\mathsf{decimal}(01) = 2$) et la troisième 11 (i.e. $\mathsf{decimal}(11) = 3$).

Exercice 217. *Donnez la matrice de parité du code 7-répétition définie de telle façon que le syndrome donne les positions de deux erreurs presque directement lorsqu'elles surviennent, et donne la position de l'erreur s'il y en a une seule.*

5.4 Co-ensembles de code (Cosets)

À partir d'un $[n,k]$–code linéaire C nous pouvons définir plusieurs codes équivalents. À chaque syndrome $s \in \{0,1\}^{n-k}$, associons le code qui contient les mots $u \in \{0,1\}^n$ tels que $s = uH^T$. Il s'agit du code obtenu en appliquant une translation fixe $a \in \{0,1\}^n$ à chaque mot de code $c \in C$. Plus précisemment :

Définition 79. *Soit C un $[n,k]$–code binaire linéaire, pour chaque $a \in \{0,1\}^n$, on définit*

$$a + C := \{a \oplus c \,|\, c \in C\} .$$

L'ensemble $a + C$ est appelé co-ensemble de C correspondant à a.

Il est clair que si C est un $[n,k,d]$–code alors $a + C$ contient 2^k chaînes de n bits, le nombre de mots de code dans C. De plus, $\Delta_{\mathsf{H}}(a + c, a + c') = \Delta_{\mathsf{H}}(c, c') \geq d$. Notez que $a + C$ n'est pas un code linéaire à proprement parler, car $a \oplus c \oplus a \oplus c' = c \oplus c' \notin a + C$ pour $c \neq c' \in C$ à moins que $a \in C$. Plus précisément,

Lemme 27. *Pour n'importe quel code linéaire C sur alphabet binaire, nous avons :*

4. Plus généralement, pour $s = s_1, s_2, \ldots, s_\ell \in \{0,1\}^\ell$ nous définissons $\mathsf{decimal}(s) := \sum_{i=1}^{\ell} 2^{\ell-i} s_i$.

1. *Chaque $b \in \{0,1\}^n$ est dans un co-ensemble, en particulier $b + C$.*
2. *Les éléments $a, b \in \{0,1\}^n$ sont dans le même co-ensemble si et seulement si $a \oplus b \in C$.*
3. *$\#(a + C) = \#C = 2^k$.*

Démonstration. L'item 1 est une conséquence du fait que C est linéaire et donc, il contient $0^n \in C$. Nous en concluons que $b \in b + C$. L'item 2 est vérifié en observant que $a, b \in a + C$ implique que $a \oplus c = b \oplus c'$ et donc que $a \oplus b \in C$ par la linéarité de C. Dans l'autre direction, il est clair que si a, b sont tels que $a \oplus b \in C$ alors $a \oplus c = b$ pour un $c \in C$. Il en résulte que $(a \oplus c) + C = a + C$, par linéarité. L'item 3 est une conséquence du fait que $a \oplus c = a \oplus c'$ si et seulement si $c = c'$. □

Le prochain lemme montre que n'importe quelle paire de co-ensembles d'un $[n,k]$–code linéaire C représente le même co-ensemble ou sont complètement disjoints. Ceci implique que l'ensemble des co-ensembles de C pavent $\{0,1\}^n$ parfaitement.

Lemme 28. *Deux co-ensembles d'un code linéaire C sont disjoints ou coïncident. L'ensemble des co-ensembles distincts de C pavent uniformément $\{0,1\}^n$:*

$$\{0,1\}^n = C \cup (a_1 + C) \cup (a_2 + C) \cup ... \cup (a_r + C) \ , \tag{5.5}$$

avec $r = 2^{n-k}$. De plus, pour chaque $y, z \in \{0,1\}^n$ et $c \neq c' \in C$,

$$y \oplus c = z \oplus c' \Leftrightarrow yH^T = zH^T \ . \tag{5.6}$$

Démonstration. L'exerice 218 vous demande de montrer que deux co-ensembles sont ou bien disjoints ou coïncident. L'équation (5.5) est une conséquence directe du lemme 27. Pour l'équation (5.6), observez que :

$$y \oplus c = z \oplus c' \Rightarrow (y \oplus c)H^T = (z \oplus c')H^T \Rightarrow yH^T = zH^T \ .$$

Dans l'autre direction, $y \oplus c \neq z \oplus c'$ pour chaque $c, c' \in C$ implique que $y \oplus z \notin C$. Nous avons donc que $yH^T \neq zH^T$. □

Exercice 218. *Montrez que deux co-ensembles d'un code linéaire C sont disjoints ou coïncident. Montrez finalement (5.5).*

Les co-ensembles d'un $[n,k]$–code C peuvent être utilisés pour réaliser la table de décodage mentionnée à la sect. 5.2.1. Pour chaque co-ensemble $a+C$, nous élisons comme représentant le mot $e \in \{0,1\}^n$ de plus faible poids de Hamming tel que $e \in a+C$. Ce représentant est appelé *meneur du co-ensemble $a + C$*. La table de décodage contient alors, pour chaque syndrome $s \in \{0,1\}^{n-k}$, le meneur $e(s) \in \{0,1\}^n$ du co-ensemble étiqueté par s. Lorsque le mot $\hat{c} \in \{0,1\}^n$ est reçu, le décodage des erreurs consiste simplement à calculer $c' = \hat{c} \oplus e(\hat{c}H^T) \in C$. Le mot c' est le résultat du décodage au plus proche de $\hat{c}$.

Les co-ensembles d'un code C peuvent générer une famille de codes lorsque nous considérons les co-ensembles qui correspondent à des mots de code dans un code linéaire C' qui contient C.

Définition 80. *Soit C_1, C_2 deux codes linéaires tels que $C_2 \subseteq C_1$. On définit $C_1/C_2 = \{c + C_2 | c \in C_1\}$, l'ensemble des co-ensembles de C_2 dans C_1.*

Les co-ensembles d'un code linéaire C_2 dans C_1 pavent C_1 comme l'ensemble des co-ensembles de C_2 dans $\{0,1\}^n$ pavent $\{0,1\}^n$ (lemme 28). En particulier,

Lemme 29. *Le nombre de co-ensembles distincts pour C_1/C_2 est $\frac{\#C_1}{\#C_2}$.*

Démonstration. Par le lemme 27, pour chaque $c \in C_1$ et $c' \in C_2$ $c \oplus c' \in C_1$ et chaque $c \in C_1$ est tel que $c \in c + C_2$, nous avons donc que $\bigcup_{X \in C_1/C_2} X = C_1$. De plus, chaque $X \in C_1$ satisfait $\#X = \#C_2$. Le lemme 28 nous montre que $y + C_2 = z + C_2$ ssi $yH_2^T = zH_2^T$ où H_2 est la matrice de parité pour C_2 et $y + C_2 \cap z + C_2 = \emptyset$ si $yH_2^T \neq zH_2^T$. Il en résulte que le nombre de co-ensembles distincts de C_2 dans C_1 est $\#C_1/\#C_2$. □

Les codes duaux nous serons très utiles pour définir une famille de codes linéaires quantiques au prochain chapitre :

Définition 81. *Soit C un $[n,k,d]$–code linéaire binaire. Le* code dual $C^\perp$ de C *est*

$$C^\perp := \{x \in \{0,1\}^n \,|\, (\forall c \in C)[\langle x, c\rangle = 0]\} \ .$$

Nous avons par construction que

$$(C^\perp)^\perp = C \ . \tag{5.7}$$

Exercice 219. *Montrez l'équation (5.7).*

De plus, soit $(h_1, h_2, \ldots, h_n)$ le vecteur contenant les colonnes de la matrice de parité H pour C. Puisque pour chaque $c \in C$,

$$cH^T = 0 \Rightarrow \sum_{i:c_i=1} h_i = 0 \ .$$

$C^\perp$ est donc le code généré par les colonnes de H. Autrement dit, H est la matrice génératrice de $C^\perp$. De la même façon, nous concluons que G est la matrice de parité pour $C^\perp$. En effet, $c'G^T = 0$ ssi $c' \in C^\perp$. Nous avons alors que pour chaque $x \in \{0,1\}^k$ et $z \in \{0,1\}^{n-k}$,

$$xGH^T z^T = (xG)(zH)^T = \langle xG, zH\rangle = 0 \ ,$$

puisque $zH \in C^\perp$ et $xG \in C$. Le dual de C est *l'annihilateur de C* par rapport au produit scalaire binaire. Le lemme suivant en est une conséquence.

Lemme 30. *Pour C un $[n,k]$–code, nous avons*

$$\dim(C) + \dim(C^\perp) = n \ .$$

Démonstration. Soit C un $[n,k]$–code avec matrice génératrice G et matrice de parité H. Le sous-espace de $GF(2)^n$ généré par C est tel que $\dim(C) = \dim(G) = k$. Le sous-espace de $\{0,1\}^n$ généré par $C^\perp$ est tel que $\dim(C^\perp) = \dim(H) = n - k$. Nous avons bien $\dim(C) + \dim(C^\perp) = n$. □

Définition 82. *C est* faiblement auto-dual *si* $C \subset C^\perp$ *et C est* auto-dual *si* $C = C^\perp$.

Les observations suivantes sont faciles à vérifier.

Exercice 220. *Montrez que le code à n–répétition est faiblement auto-dual si* $n = 0 \mod 2$ *et auto-dual si* $n = 2$.

Le prochain exercice vous demande de montrer que les codes auto-duaux ont un taux de transmission $\frac{k}{n} = \frac{1}{2}$.

Exercice 221. *Montrez qu'un* $[n,k]$*–code C est auto-dual s'il est faiblement auto-dual et* $k = \frac{n}{2}$.

La propriété suivante nous sera utile lorsque nous introduirons les codes correcteurs quantiques de type CSS. Ces codes seront construits à l'aide de codes linéaires faiblement auto-duaux.

Lemme 31. *Soit C un* $[n,k]$*–code linéaire. Pour chaque* $x \in C^\perp$ *et* $z \notin C^\perp$*, nous avons que*

$$\sum_{y \in C} (-1)^{\langle x,y \rangle} = 2^k = \#C \quad et \quad \sum_{y \in C} (-1)^{\langle z,y \rangle} = 0 \ . \tag{5.8}$$

Démonstration. La première égalité (5.8) est trivialement observée par définition du code dual $C^\perp$. D'autre part, pour $z \notin C^\perp$, nous avons

$$\sum_{y \in C} (-1)^{\langle z,y \rangle} = \sum_{w \in \{0,1\}^k} (-1)^{z G^T w^T} = \sum_{w \in \{0,1\}^k} (-1)^{\langle s,w \rangle} \ ,$$

avec $s \neq 0 \in \{0,1\}^k$ puisque G est la matrice de parité pour $C^\perp$ et $z \notin C^\perp$. Il est maintenant facile de constater que $\sum_{w \in \{0,1\}^k} (-1)^{\langle s,w \rangle} = 0$ puisqu'exactement 2^{k-1} chaînes $v \in \{0,1\}^k$ sont telles que $\langle s, v \rangle = 0$ et 2^{k-1} chaînes $v' \in \{0,1\}^k$ sont telles que $\langle s, v' \rangle = 1$. □

5.5 Code de Hamming

Le code suivant permet de corriger une erreur sur 7 bits en transmettant 4 bits d'information. Le code est donc un $[7,4,3]$–code et est appelé *code de Hamming*. La matrice génératrice du code de Hamming sous forme systématique est la suivante :

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} .$$

Le code est donc sous forme systématique. Sa matrice de parité, quant à elle, est :

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} .$$

Cette forme pour H permet de facilement recouvrer l'erreur produite. En effet, le syndrome contient la position de l'erreur écrite en binaire, car chaque colonne contient son numéro écrit en binaire.

Exemple 1. *Soit $x = 0110$ un message de 4 bits à transmettre. Le mot de code de Hamming associé à x est $c = xG = 0110011$. Supposons que durant la transmission de c, une erreur en position 3 est produite. Le receveur obtiendra $c' = 0100011$. Comme nous l'avons vu précédemment, le vecteur d'erreur $e = 0010000$ produira un syndrome $s = c'H^T$ égal à la 3ième colonne de H, $s = 011$. En interprétant s comme un nombre écrit en binaire* $\mathsf{decimal}(s)$ *dont le bit le plus significatif est s_1, nous obtenons* $\mathsf{decimal}(s) = 3$, *qui est bien tel que $e_3 = 1$.*

Regardons maintenant le dual $C^\perp$ de C. On observe que $C^\perp$ est le sous-espace généré par les rangées de H. De plus, chacune des rangées de H est obtenue par une combinaison linéaire des rangées de G. En effet, soient r_1, r_2, r_3 les rangées de H et g_1, g_2, g_3, g_4 les rangées de G. Nous avons :

$$r_1 = g_4, r_2 = g_2 \oplus g_3 \text{ et } r_3 = g_1 \oplus g_3 \ ,$$

et $C^\perp \subset C$ est donc faiblement auto dual, un $[7, 3, d]$–code avec $d \geq 3$.

Exercice 222. *Assurez de comprendre pourquoi si C est le $[7, 4, 3]$–code de Hamming alors $C^\perp$ est un $[7, 3, d]$–code avec $d \geq 3$. Trouvez la valeur de d pour $C^\perp$.*

Exercice 223. *Supposons une transmission d'un mot du code de Hamming sur un canal symétrique binaire avec paramètre $0 \leq p < \frac{1}{2}$. Donnez la valeur maximale de p pour que la transmission résulte en 4 bits décodés qui soit sans erreur avec probabilité moindre que s'ils étaient envoyés sans codage. Comparez avec votre réponse à l'exercice 214.*

5.6 Bornes sur les codes correcteurs

Une question naturelle peut maintenant être posée, quel est le taux de transmission qui puisse être atteint par un code correcteur linéaire binaire de distance d ? Plusieurs bornes sont connues pour répondre à cette question. Nous introduisons deux bornes, la borne de Hamming et la borne de Varshamov-Gilbert. La première donne un taux de transmission maximal qui puisse être atteint par un code binaire d'une certaine distance minimum. La seconde est une borne sur le taux transmission atteignable par un code binaire linéaire.

Considérez en premier lieu un mot $c \in \{0, 1\}^n$. Combien y-a-t-il de mots à distance de Hamming moindre que d de c ? Le prochain lemme, donné sans preuve, répond à cette question par un expression qui fait appel à l'entropie de Shannon de la distribution de Bernoulli avec paramètre $p = d/n$.

Lemme 32 (inégalité de la tête des coefficients binomiaux)**.** *Soient $\lambda \leq \frac{1}{2}$ et $n \in \mathbb{N}^{>0}$ tels que $\lambda n \in \mathbb{N}$. Alors,*

$$\frac{2^{nH_2(\lambda, 1-\lambda)}}{\sqrt{8n\lambda(1-\lambda)}} \leq \sum_{i=0}^{\lambda n} \binom{n}{i} \leq 2^{n\boldsymbol{H}_2(\lambda, 1-\lambda)} \ .$$

Voici la borne de Hamming pour les codes binaires. Elle établit le taux de transmission maximal qui permette un code de distance minimum donnée. La borne de Hamming est également appelée *borne de paquetage de sphères* pour les codes binaires.

Lemme 33 (borne de Hamming)**.** *Un $[n,k,d]$–code binaire (pas nécessairement linéaire) doit satisfaire*

$$\frac{k}{n} \leq 1 - \boldsymbol{H}_2\left(\frac{d}{2n}, 1 - \frac{d}{2n}\right) \ , \tag{5.9}$$

lorsque $n \to \infty$.

Démonstration. Soit C un $[n,k,d]$–code et posons $\lambda := d/2n$. Autour de $c \in C$, nous pouvons définir une sphère $S_\lambda(c) := \{x \in \{0,1\}^n | \Delta_{\mathsf{H}}(x,c) \leq \frac{d}{2}\}$ qui doit être telle que $S_\lambda(c) \cap S_\lambda(c') = 0$ pour toute paire de mots de code $c \neq c' \in C$. Nous avons,

$$\begin{aligned} 2^n &\geq \sum_{c \in C} \# S_\lambda(c) \\ &\geq \sum_{c \in C} \frac{2^{n\boldsymbol{H}_2(\lambda, 1-\lambda)}}{\sqrt{8n\lambda(1-\lambda)}} \qquad (5.10) \\ &= 2^k \left(\frac{2^{n\boldsymbol{H}_2(\lambda,1-\lambda)}}{\sqrt{8n\lambda(1-\lambda)}}\right) \ , \qquad (5.11) \end{aligned}$$

où (5.10) est une application du lemme 32. Appliquer $\lg(\cdot)$ à droite et à gauche de l'inégalité (5.11) montre ce que nous cherchons :

$$1 - \boldsymbol{H}_2(\lambda, 1-\lambda) \geq \frac{k}{n} - \frac{\lg(8n\lambda(1-\lambda))}{2n} \to \frac{k}{n} \ .$$

□

Un $[n,k,d]$–code qui corrige pn erreurs pour $0 \leq p \leq \frac{1}{2}$ doit satisfaire $d \geq 2pn$. Le lemme 33 répète ce que le théorème de codage sujet au bruit de Shannon énonce (théorème 28) dans le cas du canal symétrique binaire,

$$\frac{k}{n} \leq 1 - \boldsymbol{H}_2(p, 1-p) = C(\mathsf{C}_{\text{bin}}(p)) \ , \tag{5.12}$$

où $C(\mathsf{C}_{\text{bin}}(p))$ est la capacité du canal symétrique binaire (définition 68). La borne de Hamming ne contraint pas le taux de transmission $\frac{k}{n}$ atteignable par un $[n,k,d]$–code binaire par bloc de *distance minimum relative* $\frac{d}{n}$ au delà de la contrainte exprimée par le théorème de codage sujet au bruit de Shannon. La courbe de gauche de la Fig. 5.1 représente donc à la fois la borne de Hamming pour les $[n,k,d]$–codes binaires et la capacité du canal symétrique binaire avec paramètre $p = \frac{d}{n}$.

Plus intéressant est le fait que des codes linéaires existent qui permettent de corriger n'importe quelle fraction constante $p \leq \frac{1}{4}$ d'erreur, pour n suffisamment grand, à un taux de

transmission $\frac{k}{n} = 1 - \boldsymbol{H}_2(2p, 1-2p)$. Ce résultat est appelé la borne de Gilbert-Varshamov. Plus précisemment, elle établit qu'un code aléatoire obtenu en tirant uniformément et aléatoirement chaque entrée (chaque bit) de la matrice génératrice $G \in \mathcal{L}_{k,n}(GF(2))$ pour $k \approx n(1 - H_2(2p, 1-2p))$ aura distance minimum $d \approx 2pn$ avec probabilité essentiellement 1. La borne de Gilbert-Varshamov montre que des codes linéaires existent dont le taux de transmission n'est pas *trop loin* de la borne supérieure établie par le théorème du codage résistant au bruit, exprimée à l'équation (5.12), pour le cas spécial du canal symétrique binaire. De son côté, la preuve de Shannon (théorème 28) montrait bien que des codes aléatoires qui satisfont (5.12) existent, mais ceux-ci ne sont pas linéaires et n'ont pas nécessairement la distance minimum suffisante pour toujours corriger les erreurs de transmission. Le codage de Shannon permet de décoder sans erreur la plupart du temps (en fait, presque tout le temps) en autant que le code soit choisi aléatoirement à chaque transmission. Les codes linéaires promis par la borne de Gilbert-Varshamov peuvent être fixée une fois pour toute et vont toujours permettre de corriger $\lfloor \frac{d-1}{2} \rfloor$ erreurs.

Théorème 38 (borne de Gilbert-Varshamov)**.** *Il existe un $[n, k, d]$–code linéaire binaire pourvu que*

$$\sum_{i=0}^{d-2} \binom{n}{i} < 2^{n-k} \quad . \tag{5.13}$$

Démonstration. Par le théorème 37, il s'agit de construire une matrice de parité H pour laquelle aucun sous-ensemble des colonnes de taille $d-1$ ne contient des colonnes linéairement dépendantes. Il n'est pas trop difficile de montrer que ceci est possible lorsque (5.13) est satisfaite. □

La borne de Gilbert-Varshamov peut être exprimée d'une façon un peu plus simple en utilisant l'inégalité 32 :

Théorème 39. *Pour $0 \leq \delta \leq \frac{1}{2}$, $0 < \varepsilon \leq 1 - \boldsymbol{H}_2(\delta, 1-\delta)$ et n suffisamment grand, il existe un $[n, k, d]$–code binaire linéaire qui satisfait $\frac{k}{n} \geq 1 - \boldsymbol{H}_2(\delta, 1-\delta) - \varepsilon$ et $\frac{d}{n} \geq \delta$.*

Exercice 224. *Montrez comment obtenir le théorème 39 à partir du théorème 38.*

L'existence de codes qui satisfont aux conditions exprimées aux théorèmes 38 et 39 est établie en montrant que des codes aléatoires ont une probabilité essentiellement 1 de les satisfaire. La borne de Gilbert-Varshamov grarantit l'existence de codes qui satisfont (5.14) mais ne permet pas de les obtenir pour sûre. Trouver un code qui atteint la borne de Gilbert-Varshamov pour un n arbitraire est une tâche que nous ne savons pas résoudre efficacement en général. Nous pouvons seulement établir qu'un code aléatoire satisfait presque toujours le borne de Gilbert-Varshamov.

Exercice 225. *Montrez qu'il existe des codes binaires linéaires qui permettent de corriger presque toujours les erreurs produites sur un cancal symétrique binaire avec paramètre p pour lesquels le taux de transmission $\frac{k}{n}$ est meilleur que la moitié de la capacité du canal.*

Les codes aléatoires ont un important problème même si nous étions assuré de leur distance minimale, nous ne savons pas les décoder efficacement. Heureusement, certaines applications en cryptographie ne demandent pas aux codes correcteurs d'être efficacement décodables.

Sur un canal symétrique binaire avec paramètre p, un code linéaire avec distance minimum $d > 2pn$ est suffisant pour corriger les erreurs de transmission dans presque tous les cas. La borne de Gilbert-Varshamov nous promet que des $[n,k,2pn]$–codes binaires linéaires existent pourvu que $\frac{n-k}{n} > n\boldsymbol{H}_2(2p, 1-2p)$, car exprimé dans les termes de l'énoncé du théorème 33, un code linéaire capable de corriger $d \approx pn$ erreurs ($p \leq \frac{1}{4}$) existe tel que

$$\frac{k}{n} < 1 - \boldsymbol{H}_2\left(\frac{d}{n}, 1 - \frac{d}{n}\right) \approx 1 - \boldsymbol{H}_2(2p, 1-2p) \quad . \tag{5.14}$$

Exercice 226. *Expliquez pourquoi $p \leq \frac{1}{4}$ à l'équation (5.14).*

La borne de Gilbert-Varshamov peut être spécialisée pour les codes faiblement auto-duaux tels qu'introduits à la définition 82. Cette forme de la borne nous sera utile au prochain chapitre lorsque nous verrons les codes correcteurs quantiques du type CSS, plus particulièrement lorsque nous les utiliserons pour prouver la sécurité d'un protocole quantique de distribution de clé. La borne de Gilbert-Varshamov pour les codes faiblement auto-duaux nous permet de garantir la sécurité du protocole pour tout *taux d'erreur quantique* $p \lessapprox 0.055$. Nous verrons pourquoi par la suite. Le preuve de cette version de la borne de Gilbert-Varshamov n'est pas à notre programme. Elle est dûe à MacWilliams, Sloane et Thompson(1972).

Théorème 40 (Gilbert-Varshamov pour codes faiblement auto-duaux)**.** *Pour $p \in [0, \frac{1}{4}]$, un $[n,k,2pn]$–code faiblement auto-dual C existe pour $n \to \infty$ pourvu que*

$$\frac{k}{n} < 1 - \boldsymbol{H}_2(2p, 1-2p) \quad .$$

Les applications que nous allons considérer lorsque nous introduirons les codes quantiques du type CSS nous demanderons de construire des codes à partir de deux codes classiques, un $[n,k_1]$–code C_1 et un $[n,k_2]$–code C_2 tels que $C_2 \subseteq C_1$ et C_1 et $C_2^\perp$ ont distance minimum plus grande que d. Un code faiblement auto-dual a cette propriété en choisissant $C_1 = C^\perp$ et $C_2 = C$ pour $C^\perp$ de distance minimum d. La borne de Gilbert-Varshamov ne permet pas de garantir que $C_1 = C^\perp$ a distance minimum d, nous verrons plus tard que c'est bien le cas. Nous avons alors que $C_2 \subseteq C_1$ pour $C_1 = C_2^\perp = C^\perp$ et C_1 et $C_2^\perp$ ont distance minimum d, par construction.

Les deux bornes sur les codes binaires que nous venons d'établir sont représentées à la Fig. 5.1. La courbe de droite est la borne de Hamming et celle de gauche est la borne de Gilbert-Varshamov. La première nous donne une limite sur le taux transmission du meilleur code pour une distance minimum donnée et la seconde nous donne les taux transmission atteignables par des codes linéaires d'une distance minimum donnée.

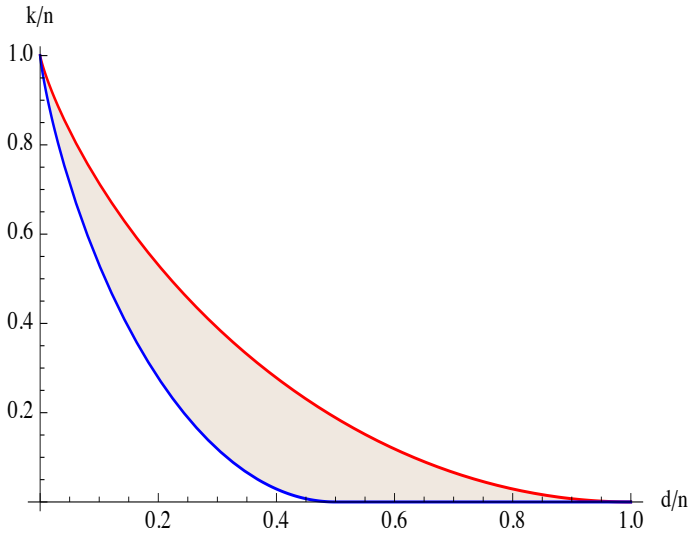

FIGURE 5.1 – Les bornes de Gilbert-Varshamov(bleu) et de Hamming(rouge) pour les codes linéaires. L'abscisse représente $\frac{d}{n}$ et l'ordonnée le taux de transmission $\frac{k}{n}$.

5.6.1 Erreurs d'effacement

Un *canal à effacement*indexcanal a effacement@canal à effacement est un canal qui remplace certains symboles par un symbole spécial qui ne fait pas partie de l'alphabet et qui est appelé symbole d'effacement. Par exemple, une mot de code binaire $(0,1,1,0,0)$ qui transit sur un canal à effacement pourraitêtre reçu comme $(0,*,1,0,*)$ où '$*$' est un symbole d'effacement. Les codes correcteurs peuvent évidemment être utilisés pour corriger les effacements. Une façon consiste à remplacer les effacements par des bits à 0 par exemple et d'utiliser la correction d'erreur du code de façon standard. On peut faire beaucoup mieux cependant, car savoir où les erreurs de transmission sont disposées ne peut qu'aider à la correction d'erreur. En fait, il n'est pas trop difficile de voir qu'un $[n,k,d]$–code permet de corriger jusqu'à $d-1$ effacements tandis qu'il ne peut corriger que $\lfloor \frac{d-1}{2} \rfloor$ renversements de bit.

Exercice 227. *Montrez qu'un* $[n,k,d]$*–code peut corriger* $d-1$ *effacements.*

5.6.2 Correction efficace des erreurs

En plus d'avoir un code avec un bon taux de transmission, les applications demandent la plupart du temps que la correction d'erreur puisse être effectuée efficacement. Ce n'est pas le cas pour les codes aléatoires. La correction d'erreurs pour ces code ne se fait pas en temps polynômial à moins que $P = NP$! En général, pour obtenir un code qui puisse être décodé efficacement, on doit sacrifier le taux de transmission atteignable pour un taux d'erreur donné. Les codes qui sont décodables efficacement et qui permettent de corriger une fraction constante d'erreurs à proximité de la borne de Gilbert-Varshamov sont rares, spécialement

pour les codes binaires. Ils doivent intégrer une structure algébrique supplémentaire qui permette d'effectuer le décodage au plus proche dans un espace de taille exponentielle en n.

5.7 Erreurs quantiques

À la sect. 4.1, nous avons vu plusieurs canaux de communication qui modifient d'une façon ou d'une autre les qubits qui y sont transmis. Les canaux de renversement de bit, de renversement de phase, de renversement de Pauli et dépolarisant peuvent tous être exprimés par l'action d'un opérateur de Pauli sur le qubit transmis avec une certaine probabilité.

Nous verrons plus loin que les erreurs de Pauli sont très importantes en théorie des codes correcteurs quantiques. Il nous sera utile de voir le bruit appliqué à un état $|\psi\rangle \in (\mathbb{C}^2)^{\otimes n}$ consituté de n qubits transmis sur un canal par une séquence d'opérateurs de Pauli. Pour $e \in \{0,1\}^n$ une chaîne de bits et $U \in \mathrm{U}_2(\mathbb{C})$ une transformation unitaire sur un qubit, nous écrivons

$$U^{\otimes e} := U^{e_1} \otimes U^{e_2} \otimes \ldots \otimes U^{e_n} \ ,$$

pour la transformation U appliquée seulement aux qubits aux positions $1 \leq i \leq n$ tels que $e_i = 1$. En particulier, $X^{\otimes e}, Z^{\otimes e}$ et $Y^{\otimes e} := X^{\otimes e} Z^{\otimes e}$ indiquent des renversements de bit, de phase et de bit et de phase aux positions i telles que $e_i = 1$ respectivement. Notez que la transformation appliquée pendant la transmission de n qubits sur le canal de renversement de Pauli peut toujours être donnée par deux chaînes d'erreurs $e^x \in \{0,1\}^n$ et $e^z \in \{0,1\}^n$ pour désigner les erreurs $X^{\otimes e^x} Z^{\otimes e^z}$.

Un cas spécial mérite notre attention, le cas d'un opérateur de Pauli appliqué sur une seule position $1 \leq i \leq n$ parmi n. Nous utilisons dans ce cas une notation différente de celle donnée plus haut :

Définition 83. *Pour $n \in \mathbb{N}^*$, nous dénoterons par $X_i, Y_i, Z_i \in \mathrm{U}_n(\mathbb{C})$ pour $1 \leq i \leq n$, les transformations de Pauli appliquées au qubit en position i d'un système de n qubits :*

$$X_i := \underbrace{\mathbb{1}_2 \otimes \ldots \mathbb{1}_2}_{i-1 \text{ fois}} \otimes X \otimes \underbrace{\mathbb{1}_2 \otimes \ldots \mathbb{1}_2}_{n-i \text{ fois}} \ , \ Z_i := \underbrace{\mathbb{1}_2 \otimes \ldots \mathbb{1}_2}_{i-1 \text{ fois}} \otimes Z \otimes \underbrace{\mathbb{1}_2 \otimes \ldots \mathbb{1}_2}_{n-i \text{ fois}} \text{ et } \ Y_i := X_i Z_i \ .$$

La transmission de n qubits dans l'état $|\psi\rangle \in (\mathbb{C}^2)^{\otimes n}$ sur un canal de renversement de bit consiste en l'opération quantique :

$$\mathfrak{X}_p(|\psi\rangle\langle\psi|) = \sum_{e \in \{0,1\}^n} p^{\mathrm{w}(e)} (1-p)^{n-\mathrm{w}(e)} X^{\otimes e} |\psi\rangle\langle\psi| X^{\otimes e} \ . \tag{5.15}$$

La même chose pour le canal de renversement de phase :

$$\mathfrak{Z}_p(|\psi\rangle\langle\psi|) = \sum_{e \in \{0,1\}^n} p^{\mathrm{w}(e)} (1-p)^{n-\mathrm{w}(e)} Z^{\otimes e} |\psi\rangle\langle\psi| Z^{\otimes e} \ . \tag{5.16}$$

Exercice 228. *Donnez une représentation semblable à celles des équations (5.15) et (5.16) pour le canal de renversement de Pauli $\mathfrak{P}_p(\cdot)$.*

5.8 L'impossibilité de cloner un état quantique

Nous montrons ici que cloner (ou copier) un état quantique arbitraire n'est pas possible par une opération permise par la mécanique quantique. Les conséquences sont multiples, nous y reviendrons.

Lemme 34. *Il n'existe aucune opération CPCT* $\mathfrak{C} : L(\mathcal{A}) \to L(\mathcal{B} \otimes \mathcal{B}')$ *avec* $\mathcal{A} \approx \mathcal{B} \approx \mathcal{B}'$ *telle que pour chaque* $|\psi\rangle \in \mathcal{A}$,

$$\mathfrak{C}(|\psi\rangle\langle\psi|) = |\psi\rangle\langle\psi| \otimes |\psi\rangle\langle\psi| \ .$$

Démonstration. Soit $U \in L(\mathcal{A}, \mathcal{B} \otimes \mathcal{B}' \otimes \mathcal{E})$ une isométrie telle que pour chaque $\rho \in \mathrm{D}(\mathcal{A})$ nous avons

$$\mathfrak{C}(\rho) = \mathrm{tr}_{\mathcal{E}}(U\rho U^*) \ . \tag{5.17}$$

Soit $|\psi\rangle, |\phi\rangle \in \mathcal{A}$ deux états purs arbitraires. En supposant l'existence de U, nous obtenons :

$$\begin{aligned} \langle\phi|\psi\rangle &= \langle\phi|U^*U|\psi\rangle \\ &= \langle\phi'|\langle\phi|\langle\phi|\psi\rangle|\psi\rangle|\psi'\rangle \\ &= \langle\phi|\psi\rangle^2\langle\phi'|\psi'\rangle, \end{aligned} \tag{5.18}$$

et (5.18) n'est pas possible, car pour $0 < |\langle\phi|\psi\rangle| < 1$ nous obtenons $\langle\phi|\psi\rangle \neq \langle\phi|\psi\rangle^2\langle\phi'|\psi'\rangle$. Nous concluons que l'opération CPCT $\mathfrak{C}$ n'existe pas. □

Exercice 229. *Considérez une généralisation du lemme 34 aux* cloneurs approximatifs. *Nous dirons d'un cloneur* $\mathfrak{C} : L(\mathcal{A}) \to L(\mathcal{B} \otimes \mathcal{B}')$ *qu'il est un* $(1-\delta)$*–cloneur si pour chaque* $|\psi\rangle \in \mathcal{A}$, $\Delta(\mathfrak{C}(|\psi\rangle\langle\psi|), |\psi\rangle\langle\psi| \otimes |\psi\rangle\langle\psi|) \leq \delta$. *Donnez une valeur* $\delta_0 > 0$ *qui soit telle qu'aucun* $(1-\delta_0)$*–cloneur n'existe.*

Exercice 230. *Construisez un* $\frac{1}{2}$*–cloneur* $\mathfrak{C} : \mathcal{L}_2(\mathbb{C}) \to \mathcal{L}_4(\mathbb{C})$ *tel que défini à l'exercice 229.*

5.9 Code quantique contre le renversement de bit

Le code 3–répétition peut être utilisé dans le monde quantique pour corriger une erreur produite durant une transmission sur un canal de renversement de bit. Un renversement de bit est le produit de l'application de l'opérateur de Pauli X sur une position (un qubit) du *mot de code quantique*. Ce code, décrit par la suite, est appelé *code 3–répétition quantique contre le renversement de bit*. Le code 3–répétition quantique sera construit un peu différemment de sa version classique, car copier des états quantiques est impossible en général. Il est toutefois possible de *copier* un qubit dans la base canonique (base de calcul), c'est-à-dire qu'il est possible et même facile de copier un qubit trois fois si celui-ci est dans un état de la base de calcul.

C'est de cette façon que nous allons procéder. Considérez l'encodage d'un qubit réalisé par l'isométrie $U_3 \in \mathcal{L}_{2,8}(\mathbb{C})$ qui produit 3 copies d'un qubit dans la base de calcul :

$$|0\rangle \quad \mapsto \quad |0_L\rangle := |000\rangle$$

$$|1\rangle \quad \mapsto \quad |1_L\rangle := |111\rangle \ .$$

Maintenant, considérez un état pur $|\psi\rangle \in \mathbb{C}^2$ arbitraire :

$$\big|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \ , \ |\alpha|^2 + |\beta|^2 = 1 \ .$$

L'encodage de $|\psi\rangle$ par l'isométrie U_3 est :

$$\big|\psi_L\rangle := U_3\big|\psi\rangle = \alpha|000\rangle + \beta|111\rangle = \alpha|0_L\rangle + \beta|1_L\rangle \ .$$

Il est clair que le projecteur

$$P_0 := |000\rangle\langle 000| + |111\rangle\langle 111| \tag{5.19}$$

n'affecte pas les états $|\psi_L\rangle$ du code, i.e. ceux qui sont produits par l'encodage U_3. Pour chaque $|\psi\rangle \in \mathbb{C}^2$ et $|\psi_L\rangle = U_3|\psi\rangle$, nous avons $|\psi_L\rangle = P_0|\psi_L\rangle$ et donc,

$$\mathrm{tr}\Big(P_0\big|\psi_L\rangle\langle\psi_L\big|\Big) = 1 \ .$$

Soit X_i pour $i \in \{1,2,3\}$, la transformation de Pauli X appliquée au qubit en position i de $|\psi_L\rangle$ (i.e. l'état $|\psi_L\rangle$ est transmis sur un canal quantique et un renversement de bit X_i lui est appliqué en transit). Le receveur obtiendra,

$$\big|\tilde{\psi}_L\big\rangle = X_i\big|\psi_L\rangle \ .$$

Considérez maintenant les projecteurs suivants :

$$P_1 := |100\rangle\langle 100| + |011\rangle\langle 011| \ , \tag{5.20}$$
$$P_2 := |010\rangle\langle 010| + |101\rangle\langle 101| \quad \text{et} \tag{5.21}$$
$$P_3 := |001\rangle\langle 001| + |110\rangle\langle 110| \ . \tag{5.22}$$

Nous avons alors $P_0 + P_1 + P_2 + P_3 = \mathbb{1}_{2^3}$ et ces projecteurs forment un observable. Il est facile de constater que :

$$\mathrm{tr}\Big(P_j\big|\tilde{\psi}_L\big\rangle\big\langle\tilde{\psi}_L\big|\Big) = \mathrm{tr}\Big(P_jX_i\big|\psi_L\rangle\langle\psi_L\big|X_i\Big) = \delta_{i,j} \ .$$

Autrement dit, l'observable $\mathsf{M}_{\mathrm{rb}} = \{P_0, P_1, P_2, P_3\}$ produit le syndrome associé au code 3–répétition. Si P_i est observé alors un renversement de bit est détecté en position i. La correction est aisée, il suffit d'appliquer X_i à $|\tilde{\psi}_L\rangle$ lorsque P_i est observé, car $X_iX_i = \mathbb{1}_{2^3}$.

Une autre façon d'exécuter la mesure du syndrome est de la séparer en deux mesures appliquées chacune sur deux qubits. Rappelons-nous comment représenter un observable par un opérateur hermitien, comme décrit à la sect. 2.9.3. Les deux mesures sont décrites par l'observable $Z \otimes Z$ sur les qubits 1 et 2 et sur les qubits 2 et 3. Notons que :

$$Z \otimes Z = |00\rangle\langle 00| + |11\rangle\langle 11| - |01\rangle\langle 01| - |10\rangle\langle 10| \ .$$

Lorsque le résultat +1 est obtenu pour les qubits 1 et 2, nous concluons que les deux qubits observés n'ont pas été affectés par un seul renversement de bit (i.e. mais les deux qubits

peuvent avoir été renversés simultanément). Lorsque le résultat -1 est obtenu alors un des deux qubits observés a subi un renversement de bit. En combinant ce résultat avec celui de la même mesure sur les qubits 2 et 3, l'endroit exact du renversement de bit peut être obtenu. Dénotons par Z_1Z_2 l'observable $Z \otimes Z$ sur les qubits 1 et 2 et Z_2Z_3 le même observable sur les qubits 2 et 3. Soit $(x, x') \in \{-1, +1\}^2$ les résultats des deux observables, x pour Z_1Z_2 et x' pour Z_2Z_3 respectivement. Nous avons :

- $(x, x') = (+1, +1)$ indique aucune erreur,
- $(x, x') = (+1, -1)$ indique une erreur X_3,
- $(x, x') = (-1, +1)$ indique une erreur X_1 et
- $(x, x') = (-1, -1)$ indique une erreur X_2.

La correction du renversement de bit peut maintenant être appliquée.

Les deux façons d'extraire le syndrome sont équivalentes. Dans certains scénarios cependant, il peut être plus facile de réaliser des mesures sur deux qubits plutôt qu'une seule mesure sur 3 qubits.

Le code que nous venons de définir permet donc de corriger la transmission de 3 qubits sur le canal de renversement de bit en autant qu'au plus un renversement se soit produit. Évidemment, la protection offerte par ce code est très limitée. Il ne peut s'agir d'un code correcteur quantique contre une erreur arbitraire puisqu'il ne peut rien contre d'autres types d'erreur comme les renversements de phase. Voyons maintenant comment détecter et corriger les renversements de phase.

Exercice 231. *Est-ce que le code* 3*–répétition contre le renversement de bit permet de corriger (par la même méthode) une rotation par un angle aléatoire* $\theta \in_R [0, \frac{\pi}{2}]$ *(inconnu du receveur) d'un seul qubit parmi les trois ? La transformation* $R(\theta)$ *qui applique une rotation par un angle* θ *est donnée par :*

$$R(\theta) := \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} .$$

5.10 Code quantique contre le renversement de phase

Considérons maintenant une erreur de transmission produite par le canal de renversement de phase. Il est facile d'utiliser une variante du code 3–répétition pour se prémunir contre ce type d'erreur. Le code résultant, que nous décrivons par la suite, sera nommé *code* 3*–répétition quantique contre le renversement de phase*. Une erreur de phase est modélisée par l'action de l'opérateur de Pauli Z sur un qubit transmis. Comme $Z = \mathsf{H}X\mathsf{H}$, où $\mathsf{H} := |+\rangle\langle 0| + |-\rangle\langle 1|$ est la transformation d'Hadamard, nous pouvons voir le renversement de phase comme un renversement de bit dans la base d'Hadamard. Ceci suggère d'encoder un qubit dans la base de calcul dans trois qubits dans la base d'Hadamard. Soit $V_3 \in \mathcal{L}_{2,8}(\mathbb{C})$ l'isométrie qui réalise cet encodage :

$$\begin{aligned} |0\rangle &\mapsto |0_L\rangle := |+++\rangle \\ |1\rangle &\mapsto |1_L\rangle := |---\rangle . \end{aligned}$$

Et un qubit dans l'état pur $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ sera encodé par

$$\big|\psi_L\rangle = V_3\big|\psi\rangle = \alpha|+++\rangle + \beta|---\rangle \ .$$

Le projecteur sur le code Q_0 est défini par :

$$Q_0 := |+++\rangle\langle+++| + |---\rangle\langle---| = \mathsf{H}^{\otimes 3} P_0 \mathsf{H}^{\otimes 3} \ .$$

Et nous avons bien que

$$\mathrm{tr}\Big(Q_0\big|\psi_L\rangle\langle\psi_L\big|\Big) = 1 \ .$$

Nous pouvons définir les projecteurs sur une erreur de phase pour chaque position :

$$\begin{aligned} Q_1 &:= |-++\rangle\langle-++| + |+--\rangle\langle+--| = \mathsf{H}^{\otimes 3} P_1 \mathsf{H}^{\otimes 3} \ , \\ Q_2 &:= |+-+\rangle\langle+-+| + |-+-\rangle\langle-+-| = \mathsf{H}^{\otimes 3} P_2 \mathsf{H}^{\otimes 3} \quad \text{et} \\ Q_3 &:= |++-\rangle\langle++-| + |--+\rangle\langle--+| = \mathsf{H}^{\otimes 3} P_3 \mathsf{H}^{\otimes 3} \ . \end{aligned}$$

Comme pour le code 3–répétition contre le renversement de bit, nous avons $Q_0 + Q_1 + Q_2 + Q_3 = \mathsf{H}^{\otimes 3}(P_0 + P_1 + P_2 + P_3)\mathsf{H}^{\otimes 3} = \mathbb{1}_{2^3}$ et pour chaque $i \in \{1,2,3\}$,

$$\mathrm{tr}\Big(Q_i Z_i\big|\psi_L\rangle\langle\psi_L\big|Z_i\Big) = 1 \ .$$

L'observable $\mathsf{M}_{\mathrm{rp}} = \{Q_0, Q_1, Q_2, Q_3\}$ permet donc de détecter l'endroit où un renversement de phase a eu lieu sur $|\psi_L\rangle$. Il s'agit du syndrome associé au code 3–répétition contre le renversement de phase. L'erreur de phase est aisément corrigée en appliquant Z_i lorsque Q_i est observé, car $Z_i Z_i = \mathbb{1}_{2^3}$.

De la même façon que pour le code contre le renversement de bit, le syndrome associé à une erreur de phase peut être obtenu en combinant deux observables appliqués sur deux qubits de l'état reçu. Ces observables sont $X_1X_2 = \mathsf{H}^{\otimes 2} Z_1 Z_2 \mathsf{H}^{\otimes 2}$ et $X_2X_3 = \mathsf{H}^{\otimes 2} Z_2 Z_3 \mathsf{H}^{\otimes 2}$. Si $(z, z') \in \{+,-\}^2$ sont les deux résultats obtenus alors :

- $(z, z') = (+1, +1)$ indique aucune erreur,
- $(z, z') = (+1, -1)$ indique une erreur Z_3,
- $(z, z') = (-1, +1)$ indique une erreur Z_1 et
- $(z, z') = (-1, -1)$ indique une erreur Z_2.

Nous venons donc de voir un code quantique qui protège contre un renversement de bit parmi 3 qubits et un autre qui protège contre un renversement de phase parmi 3 qubits. Le code de Shor présenté à la prochaine section permet de combiner ces deux codes de façon à obtenir un code qui protège contre n'importe quel renversement de Pauli.

5.11 Code de Shor

Nous savons comment détecter et corriger un renversement de bit et comment détecter et corriger un renversement de phase. Le code de Shor (1995) utilise ces deux codes correcteurs

pour en construire un qui détecte et corrige une erreur de Pauli arbitraire. Ces travaux de Shor peuvent être qualifiés de percée majeure en théorie de l'information quantique. Pour la première fois, ils permettent d'envisager la mise au point d'un ordinateur quantique qui puisse résister au bruit inhérent aux systèmes quantiques laissés à eux-même. Cette tâche pouvait même sembler impossible à l'époque. Comment ajouter de la redondance à un état quantique pour le rendre résistant au bruit lorsque copier est impossible ?

Le code de Shor encode en premier lieu le qubit avec le code 3–répétition contre le renversement de phase et ensuite encode chacun des 3 qubits résultants avec le code 3–répétition contre le renversement d'un bit. Soit $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ un qubit dans un état pur. L'encodage de $|\psi\rangle$ avec le code de Shor est $|\psi_L\rangle = (U_3 \otimes U_3 \otimes U_3)V_3|\psi\rangle$. Nous obtenons :

$$\begin{aligned}\big|\psi_L\big\rangle = &\frac{\alpha}{2\sqrt{2}}(|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \\ &+ \frac{\beta}{2\sqrt{2}}(|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \ .\end{aligned}$$

Le code de Shor encode donc 1 qubit dans 9. Montrons qu'il corrige n'importe quelle erreur de Pauli affectant un seul qubit. La détection et la correction d'erreur pour le code de Shor se fait en deux phases séparées. La première phase détecte et corrige les renversements de bit pour chaque groupe de trois qubits. Notez que plusieurs erreurs peuvent être corrigées en autant que chaque groupe de trois qubits n'est affecté que par au plus une seule erreur. L'observable pour l'obtention du syndrome de renversement de bit est obtenu en appliquant l'observable $\{P_0, P_1, P_2, P_3\}$ défini aux équations (5.19),(5.20),(5.21) et (5.22) sur chaque triplet, il s'agit de l'observable $\mathsf{M}_{\mathrm{bf}} = \{P_{x_1} \otimes P_{x_2} \otimes P_{x_3}\}_{x_1,x_2,x_3 \in \{0,1,2,3\}}$. La correction est effectuée sur le premier bloc de 3 qubits en position x_1 (si $x_1 \neq 0$), sur le deuxième bloc en position $3 + x_2$ (si $x_2 \neq 0$) et sur le troisième bloc en position $6 + x_3$ (si $x_3 \neq 0$).

La deuxième phase vise à corriger un renversement de phase. Notez que pour $1 \leq i, j, j' \leq 3$

$$Z_{3(i-1)+j}\big|\psi_L\big\rangle = Z_{3(i-1)+j'}\big|\psi_L\big\rangle \ .$$

C'est donc inutile de détecter la position d'un renversement de phase plus précisément qu'en localisant le bloc de trois qubits sur lequel un renversement a eu lieu. Soit $|G^+\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$ et $|G^-\rangle = \frac{1}{\sqrt{2}}(|000\rangle - |111\rangle)$. Le syndrome pour le renversement de phase est obtenu en mesurant avec $\{Q'_0, Q'_1, Q'_2, Q'_3\}$ où

$$\begin{aligned}
Q'_0 &= \big|G^+\rangle\langle G^+\big| \otimes \big|G^+\rangle\langle G^+\big| \otimes \big|G^+\rangle\langle G^+\big| + |G^-\rangle\langle G^-| \otimes |G^-\rangle\langle G^-| \otimes |G^-\rangle\langle G^-| \ , \\
Q'_1 &= |G^-\rangle\langle G^-| \otimes \big|G^+\rangle\langle G^+\big| \otimes \big|G^+\rangle\langle G^+\big| + \big|G^+\rangle\langle G^+\big| \otimes |G^-\rangle\langle G^-| \otimes |G^-\rangle\langle G^-| \ , \\
Q'_2 &= \big|G^+\rangle\langle G^+\big| \otimes |G^-\rangle\langle G^-| \otimes \big|G^+\rangle\langle G^+\big| + |G^-\rangle\langle G^-| \otimes \big|G^+\rangle\langle G^+\big| \otimes |G^-\rangle\langle G^-| \text{ et} \\
Q'_3 &= \big|G^+\rangle\langle G^+\big| \otimes \big|G^+\rangle\langle G^+\big| \otimes |G^-\rangle\langle G^-| + |G^-\rangle\langle G^-| \otimes |G^-\rangle\langle G^-| \otimes \big|G^+\rangle\langle G^+\big| \ .
\end{aligned}$$

Nous avons que pour $1 \leq i, j \leq 3$,

$$\mathrm{tr}\Big(Q'_0\big|\psi_L\rangle\langle\psi_L\big|\Big) = 1 \text{ et } \mathrm{tr}\Big(Q'_i Z_{3(i-1)+j}\big|\psi_L\rangle\langle\psi_L\big|Z_{3(i-1)+j}\Big) = 1 \ .$$

Si le syndrome $Q'_j, 1 \leq j \leq 3$ est obtenu alors $Z_{3(j-1)+1}$ corrigera le renversement de phase. Notez que $\{Q'_0, Q'_1, Q'_2, Q_3\}$ n'est pas un observable

Exercice 232. *Supposons que le code de Shor est utilisé pour des transmissions sur le canal de Pauli avec paramètre $0 \leq p \leq \frac{1}{2}$. Quelle est la valeur maximale de p qui permette au code de Shor de produire un qubit du côté du receveur dont la probabilité d'erreur est meilleure que p ?*

Le code de Shor détecte et corrige un renversement de bit et un renversement de phase arbitraire. En particulier, ceci signifie que le code de Shor protège également contre un renversement de bit et un renversement de phase appliqués sur le même qubit. Ceci correspond à la transformation de Pauli Y. En conclusion, nous avons un code qui corrige une erreur de Pauli arbitraire sur 9 qubits, une code qui protège contre un renversement de Pauli. Comme le montre l'exercice 167, le code de Shor protège donc également pour le même prix contre une erreur introduite par le canal dépolarisant.

Exercice 233. *Donnez une disposition pour le maximum d'erreurs X et Z que le code de Shor permettrait de corriger avec succès si la correction des renversements de bit pour chaque triplet était exécutée complètement peu importe ce qui est survenu aux triplets précédents.*

Exercice 234. *Supposons que vous disposiez d'un appareil permettant l'encodage, le décodage et la correction pour le code de Shor. Montrez comment vous pourriez utiliser cet appareil pour encoder, décoder et corriger deux erreurs de Pauli arbitraires sur 81 qubits. Généralisez en montrant comment mettre au point un code sur 9^k qubits qui permettent la correction de n'importe quelles k erreurs de Pauli. Les codes construits de cette façon sont appelés* codes concaténés

Les *codes concaténés* permettent de convertir un code par bloc qui corrige t erreurs en un code pour des blocs de taille croissante pour lequel le taux d'erreur de décodage peut tendre vers 0 exponentiellement rapidement dans la longueur du bloc. La construction utilise le code par bloc comme code interne et un code externe pour corriger les erreurs de décodage du code interne. Le code externe peut être dérivé ou non du code interne. Habituellement, le code externe est un code sur un alphabet qui contient autant de symboles que le nombre de messages qui peut être codé par le code interne. Cette technique, dûe à D. Forney en 1966, à été utilisée pour les communications spatiales dans les années 1970s.

5.12 L'universalité du bruit de Pauli

Tentons d'identifier le bruit contre lequel le code de Shor protège l'information quantique. Nous savons qu'il protège contre le canal de renversement de Pauli qui affecte un seul qubit. Peut-il faire mieux ? Nous verrons qu'il protège en fait contre n'importe quel type de bruit qui n'affecte qu'un seul qubit.

Supposons que l'environnement applique l'opération quantique $\mathfrak{E} = \{E_k\}_k$ sous la forme de Kraus. Si le mot de code quantique $|\psi_L\rangle$ est transmis sur le canal, l'état sera modifié comme :

$$\mathfrak{E}(|\psi_L\rangle\langle\psi_L|) = \sum_k E_k |\psi_L\rangle\langle\psi_L| E_k^* \ .$$

Rappelons-nous que l'interprétation physique de l'opération quantique $\mathfrak{E} = \{E_k\}_k$ est celle que nous avons vue à la sect. 3.10.2, avec probabilité $\mathrm{tr}(|\psi_L\rangle\langle\psi_L|E_k^* E_k)$ l'opération quantique appliquée est

$$\frac{E_k|\psi_L\rangle\langle\psi_L|E_k^*}{\langle\psi_L|E_k^* E_k|\psi_L\rangle} \ .$$

Supposons maintenant que l'opération $\mathfrak{E}$ affecte un seul qubit pendant la transmission de $|\psi_L\rangle$. Il en résulte que E_k affecte un seul qubit pour chaque k. Si E_k n'affecte qu'un seul qubit alors E_k agit comme l'identité sur toutes les positions sauf une, soit i_k cette position.

Puisque $\{\mathbb{1}, X, Y, Z\}$ forme une base pour les opérateurs sur un qubit (i.e. comme mentionné à la sect. 2.5), nous pouvons écrire E_k comme une combinaison linéaire aux coefficients complexes de celles-ci :

$$E_k = e_0\mathbb{1} + e_x X_{i_k} + e_y Y_{i_k} + e_z Z_{i_k} \ , \tag{5.23}$$

et

$$E_k|\psi_L\rangle = e_0|\psi_L\rangle + e_x X_{i_k}|\psi_L\rangle + e_y Y_{i_k}|\psi_L\rangle + e_z Z_{i_k}|\psi_L\rangle \ . \tag{5.24}$$

L'obtention du syndrome complet fixera l'erreur de Pauli, X_{i_k} avec probabilité $|e_x|^2/\langle\psi_L|E_k^* E_k|\psi_L\rangle$, Y_{i_k} avec probabilité $|e_y|^2/\langle\psi_L|E_k^* E_k|\psi_L\rangle$ et Z_{i_k} avec probabilité $|e_z|^2/\langle\psi_L|E_k^* E_k|\psi_L\rangle$. Cette observation est prouvée à la section suivante dans un contexte plus général.

Exercice 235. *Montrez comment appliquer le raisonnement qui mène à (5.23) et (5.24) dans le cas où E_k est garanti de n'affecter que t qubits dans le pire des cas.*

Voyons maintenant des conditions qui font que le code de Shor corrige une erreur arbitraire sur 9 qubits. Nous allons obtenir des conditions que doivent satisfaire les codes correcteurs quantiques pour qu'ils soient possible de corriger le bruit $\mathfrak{E} = \{E_i\}_i$, exprimé sous forme de Kraus. Soit $\{|\psi_\ell\rangle\}_\ell$ une base pour le sous-espace P_C généré par les mots du code.

Une condition qui doit être respectée est que pour $E_j, E_{j'} \in \{E_i\}_i$, les mots de codes $|\psi_\ell\rangle$ et $|\psi_{\ell'}\rangle$ pour $\ell \neq \ell'$ ne doivent pas être confondus par les erreurs E_j et $E_{j'}$:

$$\langle\psi_\ell|E_j^* E_{j'}|\psi_{\ell'}\rangle = 0 \ , \tag{5.25}$$

c'est-à-dire que $E_j|\psi_\ell\rangle$ et $E_{j'}|\psi_{\ell'}\rangle$ sont orthogonaux. Autrement, la correction ne pourrait pas toujours corriger les erreurs produites par l'opération quantique $\{E_i\}_i$.

Exercice 236. *Montrez que si (5.25) n'est pas satisfait alors le code n'arrive pas toujours à corriger avec succès les erreurs produites par l'opération $\{E_i\}_i$.*

Il est tentant d'ajouter une autre condition pour qu'un code correcteur permette de corriger les erreurs $\{E_i\}_i$. Le syndrome des erreurs doit être différent pour deux erreurs différentes sur le même mot de code. Autrement, comment pourrions-nous corriger ? Cette condition peut être écrite pour tout $i \neq j$ comme :

$$\langle\psi_\ell|E_i^* E_j|\psi_\ell\rangle = 0 \ . \tag{5.26}$$

Cette condition n'est cependant pas nécessaire pour corriger avec succès E_i et E_j. Le code de Shor, par exemple, corrige une erreur de phase Z_1, Z_2, Z_3 de la même façon, sans distinguer Z_1 de Z_2 et Z_3. La même remarque pour les erreurs de phase Z_4, Z_5, Z_6 et Z_7, Z_8, Z_9. Nous n'avons donc pas besoin d'identifier l'erreur précisemment pour pouvoir la corriger. La condition que nous verrons à la prochaine section pour qu'un code corrige pour $\{E_i\}_i$ est qu'il existe une façon de représenter l'opération quantique avec décomposition de Kraus $\{E_i\}_i$ par une décomposition de Kraus $\{E'_i\}_i$ pour laquelle la condition (5.26) est vérifiée (i.e. $\langle\psi_\ell|E'^*_i E'_j|\psi_\ell\rangle = 0$ pour $i \neq j$). C'est ce qui est montré à la section suivante, au théorème 42.

5.13 Conditions pour la correction d'erreur

Soit C un code quantique qui protège contre le bruit $\mathfrak{E} = \{E_i\}_i$. Supposons qu'il encode un état quantique de k qubits $|\psi\rangle \in (\mathbb{C}^2)^{\otimes k}$ dans un état quantique $|\psi_L\rangle \in (\mathbb{C}^2)^{\otimes n}$ de n qubits avec $n > k$, via une isométrie $U \in \mathcal{L}_{2^k,2^n}(\mathbb{C})$ telle que $|\psi_L\rangle := U|\psi\rangle$. Nous supposons que le bruit $\mathfrak{E}$ est CPCT, il préserve donc la trace.

L'opération qui corrige les erreurs $\mathfrak{E}$ et décode ensuite peut être vue comme une opération CPCT $\mathfrak{R} : \mathcal{L}_{2^n}(\mathbb{C}) \to \mathcal{L}_{2^k}(\mathbb{C})$ avec la propriété

$$\mathfrak{R}(\mathfrak{E}(|\psi_L\rangle\langle\psi_L|)) = |\psi\rangle\langle\psi| \ .$$

En pratique, $\mathfrak{R}$ *mesure* le syndrome et corrige les erreurs qui y sont associées avant de décoder en exécutant U^*. Pour C un code quantique, nous définissons $P_C = \sum_{x\in\{0,1\}^k} U|x\rangle\langle x|U^*$, le projecteur sur l'espace du code C.

Lemme 35. *Soit C un code quantique et soit P_C le projecteur sur celui-ci et soit $\mathfrak{R}$ l'opération CPCT qui corrige et décode les mots du code C. Soit $\mathfrak{E}$ une opération quantique CPCT avec décomposition de Kraus $\{E_i\}_i$. Si $\mathfrak{R}$ corrige $\mathfrak{E}$ alors*

$$P_C E_i^* E_{i'} P_C = \alpha_{i,i'} P_C \ , \tag{5.27}$$

où $(\alpha_{i,i'})_{i,i'}$ est une matrice hermitienne de nombres complexes.

Démonstration. Soit $\mathfrak{E} = \{E_i\}_i$ un ensemble d'erreurs CPCT qui peuvent être corrigées et décodées par $\mathfrak{R} = \{R_h\}_h$ sous forme de Kraus. Pour chaque mot de code $|\psi_L\rangle = U|\psi\rangle$, où $U \in L(\mathcal{A}, \mathcal{A}')$ est l'isométrie d'encodage du code, nous avons

$$\sum_{h,i} U R_h E_i |\psi_L\rangle\langle\psi_L| E_i^* R_h^* U^* = |\psi_L\rangle\langle\psi_L| \ . \tag{5.28}$$

Puisque $\mathfrak{E}$ et $\mathfrak{R}$ préservent la trace, (5.28) implique que pour chaque $\rho \in \mathrm{D}(\mathcal{A})$,

$$\sum_{h,i} U R_h E_i P_C \rho P_C E_i^* R_h^* U^* = P_C \rho P_C \ . \tag{5.29}$$

Nous concluons que les opérateurs de Kraus $\{UR_hE_iP_C\}_{h,i}$ définissent une opération équivalente à la seule opération $\{P_C\}$. Le théorème 22 permet [5] de conclure que chaque $UR_hE_iP_C = c_{h,i}P_C$ pour $c_{h,i} \in \mathbb{C}$ (i.e. les opérateurs d'une forme peuvent être représentés par une combinaison linéaire des opérateurs de l'autre forme). Et en prenant l'adjoint, $P_CE_i^*R_h^*U^* = c_{h,i}^*P_C$, nous obtenons

$$P_CE_i^*R_h^*U^*UR_hE_{i'}P_C = P_CE_i^*R_h^*R_hE_{i'}P_C = c_{h,i}^*c_{h,i'}P_C \ .$$

En sommant sur h, nous obtenons :

$$\sum_h P_CE_i^*R_h^*R_hE_{i'}P_C = P_CE_i^*E_{i'}P_C = \sum_h c_{h,i}^*c_{h,i'}P_C = \alpha_{i,i'}P_C \ .$$

Notons finalement que $\alpha_{i,i'} = \sum_h c_{h,i}^*c_{h,i'} = \sum_h (c_{h,i'}^*c_{h,i})^* = \alpha_{i',i}^*$ et $(\alpha_{i,i'})_{i,i'}$ est hermitien. □

Exercice 237. *Montrez que le code contre un renversement de phase satisfait la condition du lemme 35.*

Exercice 238. *Montrez que le code de Shor satisfait la condition du lemme 35 pour les erreurs* $\{X_i, Y_i, Z_i\}_{i=1}^9$.

L'équation (5.27) n'est pas seulement une condition nécessaire pour la correction des erreurs $\{E_i\}_i$, elle est également suffisante. Cette condition est appelée la *condition de correction d'erreurs quantiques*. Tout code pouvant corriger $\{E_i\}_i$ doit satisfaire (5.27) et tout code pour lequel (5.27) est satisfait peut corriger $\{E_i\}_i$. La suffisance de la condition (5.27) pour l'existence d'un code correcteur pour $\{E_i\}_i$ est exprimée au théorème suivant qui est donné sans preuve.

Lemme 36. *Soit C un code quantique et soit P_C le projecteur sur celui-ci. Soit $\mathfrak{E}$ une opération quantique CPCT avec décomposition de Kraus $\{E_i\}_i$. Il existe une opération CPCT $\mathfrak{R}$ qui corrige et décode les mots de code modifiés par $\mathfrak{E}$ lorsque*

$$P_CE_i^*E_{i'}P_C = \alpha_{i,i'}P_C \ , \tag{5.30}$$

où $(\alpha_{i,i'})_{i,i'}$ est une matrice hermitienne de nombres complexes.

La condition exprimée aux lemmes 35 et 36 peut être vue un peu différemment. Soit $\{|\psi_\ell\rangle\}_\ell$ une base orthornormée pour le sous-espace P_C du code C qui protège contre les opérateurs de bruit $\{E_i\}_i$. La condition (5.30) implique donc que :

$$\langle\psi_\ell|P_CE_i^*E_{i'}P_C|\psi_{\ell'}\rangle = \alpha_{i,i'}\langle\psi_\ell|P_C|\psi_{\ell'}\rangle \Rightarrow \langle\psi_\ell|E_i^*E_{i'}|\psi_{\ell'}\rangle = \alpha_{i,i'}\delta_{\ell,\ell'} \ . \tag{5.31}$$

5. Le théorème 22 nous dit que si deux décompositions de Kraus $\{E_h\}_h \in 2^{L(\mathcal{A})}$ et $\{F_j\}_j \in 2^{L(\mathcal{A})}$ sont telles que pour chaque $\rho \in \mathrm{D}(\mathcal{A})$, $\sum_h E_h\rho E_h^* = \sum_j F_j\rho F_j^*$ alors chaque opérateur d'une décomposition est une combinaison linéaire des opérateurs de l'autre décomposition avec les coefficients de chacune des combinaisons formant un matrice unitaire. Le théorème demeure vrai même si $\{E_h\}_h$ et $\{F_j\}_j$ ne sont pas des opérations quantiques qui préservent la trace. C'est le cas ici, $\{P_C\}$ ne préserve pas la trace.

L'égalité de droite à l'équation (5.31) indique que pour $\ell \neq \ell'$, $\langle \psi_\ell | E_i^* E_{i'} | \psi_{\ell'} \rangle = 0$, ce qui est la condition établie à l'équation (5.25). Nous avons également que $\langle \psi_\ell | E_i^* E_i | \psi_\ell \rangle = \alpha_{i,i} = \langle \psi_{\ell'} | E_i^* E_i | \psi_{\ell'} \rangle$, et l'erreur E_i ne change pas la taille relative des états du code sur lesquels elle agit. Le théorème suivant utilise cette façon d'écrire la condition pour la correction d'erreur donnée aux lemmes 35 et 36. La condition résultante est équivalente à la condition originale.

Théorème 41. *Un qcode C, avec projecteur P_C sur le sous-espace qu'il engendre, corrige les erreurs $\mathfrak{E} = \{E_i\}_i$ si et seulement si pour chaque i, i' nous avons*

$$\left\langle \psi_\ell \middle| E_i^* E_{i'} \middle| \psi_{\ell'} \right\rangle = \alpha_{i,i'} \delta_{\ell,\ell'} \ ,$$

où $(\alpha_{i,i'})_{i,i'}$ est un opérateur hermitien et $\{|\psi_\ell\rangle\}_\ell$ est une base orthornormée pour P_C.

Nous montrons maintenant que $\mathfrak{R}$ protège contre n'importe quel bruit $\mathfrak{F}$ dont les opérateurs de Kraus $\{F_j\}_j$ sont des combinaisons linéaires des opérateurs de Kraus pour $\mathfrak{E}$. En particulier, un code qui protège contre t erreurs produites par le canal dépolarisant, corrige également t erreurs produites par le canal de Pauli et toutes les erreurs qui n'affectent pas plus de t positions.

Théorème 42. *Supposons que C est un qcode et $\mathfrak{R}$ est une opération de correction d'erreurs et décodage pour les erreurs produites par $\mathfrak{E} = \{E_i\}_i$. Soit $\mathfrak{F} = \{F_j\}_j$ une autre opération modélisant les erreurs telle que pour chaque j, $F_j = \sum_i e_{j,i} E_i$ avec $e_{j,i} \in \mathbb{C}$ alors $\mathfrak{R}$ protège contre $\mathfrak{F}$ également.*

Démonstration. Soit $|\psi_L\rangle \in \mathcal{C}$ un mot de code et $|\psi\rangle \in \mathcal{A}$ tel que $|\psi_L\rangle = U|\psi\rangle$, l'état encodé dans $|\psi_L\rangle$. Puisque $\mathfrak{R}$ corrige les erreurs $\{E_i\}_i$, le lemme 35 nous indique que

$$P_C E_i^* E_{i'} P_C = \alpha_{i,i'} P_C \ ,$$

pour $(\alpha_{i,i'})_{i,i'}$ hermitien. Par le théorème de décomposition spectrale, il existe un opérateur unitaire u tel que $u \alpha u^* = d$ avec d diagonal. Posons $\{E'_\ell\}_\ell$ avec

$$E'_\ell = \sum_i u_{i,\ell} E_i \ ,$$

est tel que $\mathfrak{E}$ est une opération CPCT avec décomposition de Kraus $\{E'_\ell\}_\ell$, car le théorème 22 nous indique que les décompositions de Kraus $\{E_i\}_i$ et $\{E'_\ell\}_\ell$ décrivent la même opération lorsque chaque E'_ℓ peut être représenté par une combinaison linéaire de $\{E_k\}_k$ avec coefficients $(u_{\ell,i})_{\ell,i}$ qui forment un opérateur unitaire. Si pour chaque j et ℓ, $F_j = \sum_i e_{j,i} E_i$ et $E'_\ell = \sum_i u_{\ell,i} E_i$ alors $F_j = \sum_\ell e'_{j,\ell} E'_\ell$ pour $e'_{j,\ell} \in \mathbb{C}$. Observons maintenant que,

$$\begin{aligned} P_C E'^*_\ell E'_{\ell'} P_C &= P_C \left(\sum_i u^*_{\ell,i} E_i^* \right) \left(\sum_{i'} u_{\ell',i'} E_{i'} \right) P_C \\ &= \sum_{i,i'} u^*_{\ell,i} u_{\ell',i'} P_C E_i^* E_{i'} P_C \end{aligned}$$

$$
\begin{aligned}
&= \sum_{i,i'} u^*_{\ell,i}\alpha_{i,i'}u_{\ell',i'}P_C \\
&= d_{\ell,\ell'}P_C \ . \qquad (5.32)
\end{aligned}
$$

L'égalité (5.32) implique que pour $\ell \neq \ell'$ et pour chaque mot de code $|\psi_L\rangle$, nous avons $\langle\psi_L|E'^*_\ell E'_{\ell'}|\psi_L\rangle = 0$, car $P_C E'^*_\ell E'_{\ell'} P_C = 0$. Autrement dit, E'_ℓ transforme $|\psi_L\rangle$ en un état orthogonal à $E'_{\ell'}|\psi_L\rangle$ pour chaque $\ell' \neq \ell$. Les deux erreurs sont parfaitement distinguables.

Nous avons,

$$
\begin{aligned}
\mathfrak{R}(\mathcal{F}(\left|\psi_L\right\rangle\!\left\langle\psi_L\right|)) &= \mathfrak{R}\left(\sum_j F_j \left|\psi_L\right\rangle\!\left\langle\psi_L\right| F_j^*\right) \\
&= \mathfrak{R}\left(\sum_j \left(\sum_\ell e'_{j,\ell}E'_\ell\right)\left|\psi_L\right\rangle\!\left\langle\psi_L\right|\left(\sum_{\ell'} e'^*_{j,\ell'}E'^*_{\ell'}\right)\right) \\
&= \sum_{j,\ell,\ell'} e'_{j,\ell}e'^*_{j,\ell'}\mathfrak{R}(E'_\ell\left|\psi_L\right\rangle\!\left\langle\psi_L\right|E'^*_{\ell'}) \ . \qquad (5.33)
\end{aligned}
$$

Pour une isométrie $W \in L(\mathcal{C}, \mathcal{A}\otimes\mathcal{X})$, nous avons $\mathfrak{R}(\rho) = \mathrm{tr}_{\mathcal{X}}(W\rho W^*)$. Nous pouvons maintenant ré-écrire (5.33) comme

$$
\begin{aligned}
\mathfrak{R}(\mathcal{F}(\left|\psi_L\right\rangle\!\left\langle\psi_L\right|)) &= \sum_{j,\ell\neq\ell'} e'_{j,\ell}e'^*_{j,\ell'}\,\mathrm{tr}_{\mathcal{X}}\left(WE'_\ell\left|\psi_L\right\rangle\!\left\langle\psi_L\right|E'^*_{\ell'}W^*\right) + \sum_{j,\ell}|e'_{j,\ell}|^2\,\mathrm{tr}_{\mathcal{X}}\left(WE'_\ell\left|\psi_L\right\rangle\!\left\langle\psi_L\right|E'^*_\ell W^*\right) \\
&= \sum_{j,\ell}|e'_{j,\ell}|^2\,\mathrm{tr}_{\mathcal{X}}\left(WE'_\ell\left|\psi_L\right\rangle\!\left\langle\psi_L\right|E'^*_\ell W^*\right) \qquad (5.34) \\
&= \sum_{j,\ell}|e'_{j,\ell}|^2\,\mathfrak{R}(E'_\ell\left|\psi_L\right\rangle\!\left\langle\psi_L\right|E'^*_\ell) \\
&= \left|\psi\right\rangle\!\left\langle\psi\right| \ , \qquad (5.35)
\end{aligned}
$$

où (5.34) est une conséquence de (5.32), les états non-normalisés $|\psi\rangle\otimes|\tilde{\varphi}_\ell\rangle = WE'_\ell|\psi_L\rangle$ et $|\psi\rangle\otimes|\tilde{\varphi}_{\ell'}\rangle = WE'_{\ell'}|\psi_L\rangle$ sont orthogonaux lorsque $\ell \neq \ell'$. L'exercice 239 vous demande de vérifier cette affirmation. Il s'en suit que $\mathrm{tr}_{\mathcal{X}}(WE'_\ell|\psi_L\rangle\!\langle\psi_L|E'^*_{\ell'}W^*) = \mathrm{tr}_{\mathcal{X}}(|\psi\rangle\!\langle\psi|\otimes|\varphi_\ell\rangle\!\langle\varphi_{\ell'}|_{\mathcal{X}}) = \langle\varphi_{\ell'}|\varphi_\ell\rangle|\psi\rangle\!\langle\psi| = 0$, car $\langle\varphi_\ell|\varphi_{\ell'}\rangle = 0$. L'équation (5.35) est une conséquence du fait que $\mathfrak{R}$ corrige E'_ℓ. □

Exercice 239. *Retour à la preuve du théorème 42. Considérez le code C de l'énoncé qui corrige pour $\mathfrak{E} = \{E'_i\}_i$, l'opérateur d'erreur $E'_\ell \in L(\mathcal{C})$ qui peut être corrigé par le code C et ensuite décodé par l'isométrie $W \in L(\mathcal{C}, \mathcal{A}\otimes\mathcal{X})$ qui réalise $\mathfrak{R}$. Nous avons alors $|\psi\rangle\otimes|\tilde{\varphi}_\ell\rangle = WE'_\ell|\psi_L\rangle$ pour $|\psi_L\rangle$ l'encodage de l'état $|\psi\rangle$ selon le code C et où $|\tilde{\varphi}_\ell\rangle$ est un état possiblement non normalisé. Montrez que pour E'_ℓ et $E'_{\ell'}$, deux erreurs distinctes, nous avons $\langle\tilde{\varphi}_\ell|\tilde{\varphi}_{\ell'}\rangle = 0$.*

Exercice 240. *Montrez qu'un code qui permet de corriger un erreur Y (mais pas une erreur X ni une erreur Z) corrige également une erreur de rotation $R(\theta)$, comme définie à l'exercice 231.*

5.14 Codes contre les erreurs arbitraires et les effacements

Les codes quantiques qui permettent de corriger n'importe quelles erreurs sur t positions sont appelés [6] *codes correcteurs qui corrigent t erreurs* (nous laisserons souvent tomber l'adjectif *correcteur* et l'utilisation du terme *code* désignera un code correcteur par la suite). Un code quantique C qui encode k qubits dans n est dit de *dimension k* et de longueur n. Cette façon de définir la dimension d'un code peut porter à confusion. En effet, la dimension du sous-espace de $(\mathbb{C}^2)^{\otimes n}$ engendré par les mots de code de C est 2^k et non k.

La distance minimum d'un code quantique est un peu différente de sa version classique. Il s'agit du nombre de positions minimum sur lesquelles des opérations permettent de passer d'un mot du code à un autre. Il s'agit du chemin le plus court entre deux mots du code déterminé par des opérations qui affectent un seul qubit, appliquées une à la fois séquentiellement. Si le code corrige t erreurs alors sa distance minimum doit satisfaire la même identité que pour les codes classiques :

$$\left\lfloor \frac{d-1}{2} \right\rfloor \geq t \ . \tag{5.36}$$

Exercice 241. *Montrez l'équation (5.36).*

Définition 84. *Un code quantique C de dimension k, de longueur n et de distance minimum d est appelé* $[[n,k,d]]$–qcode. *Lorsque la distance minimum n'est pas connue ou n'est pas utile dans le contexte, nous disons de C qu'il est un* $[[n,k]]$–qcode. *Le* taux de transmission *d'un* $[[n,k]]$*–qcode est* $\frac{k}{n}$.

Le code de Shor est un $[[9,1]]$–qcode qui corrige une erreur. Nous verrons plus loin qu'il existe des codes quantiques capables de corriger une erreur avec une taux de transmission plus élevé que le code de Shor.

Exercice 242. *Donnez la distance minimum du code de Shor. Expliquez votre réponse.*

Le prochain lemme montre qu'un code qui corrige t erreurs arbitraires sur un mot de code peut corriger $2t$ erreurs à des endroits connus. Autrement dit, un $[[n,k,d]]$–qcode permet de corriger $d-1$ erreurs à des positions connues, tandis qu'il peut corriger $\lfloor \frac{d-1}{2} \rfloor$ erreurs arbitraires. Cette propriété nous permettra d'établir que le code le plus court pour un qubit qui permet de corriger n'importe quelle erreur doit avoir longueur $n \geq 5$.

Lemme 37. *Soit C un* $[[n,k]]$*–qcode qui corrige t erreurs arbitraires. Alors, C permet de corriger* $2t$ *erreurs lorsque les positions erronées sont connues.*

Démonstration. Il s'agit de montrer qu'un code qui protège contre toutes les erreurs sur t positions satisfait la condition de correction d'erreur décrite au théorème 41 pour au plus $2t$ erreurs arbitraires aux positions connues $I \subset \{1,\ldots,n\}$. Soit P_C le projecteur sur le qcode C et soit $\{|\psi_\ell\rangle\}_\ell$ une base orthornormée pour le sous-espace défini par P_C.

6. d'une façon très originale.

Puisque C corrige t erreurs arbitraires, il permet de corriger les erreurs produites par l'opération quantique $\mathfrak{Q}$ avec décomposition de Kraus $\mathfrak{Q} := \{\sqrt{\gamma}X^{\otimes x}Z^{\otimes z}\}_{x,z\in\{0,1\}^n:\#\{i|x_i=1\wedge z_i=1\}\leq t} =: \{E_j\}_j$ où $0 < \gamma < 1$ est choisi pour que γ soit la probabilité d'une distribution uniforme sur les opérateurs de Pauli de *poids* au plus t (où le poids de $X^{\otimes x}Z^{\otimes z}$ désigne le nombre de positions i t.q. $x_i = 1 \vee z_i = 1$) . Par le théorème 41, le code C satisfait

$$\langle\psi_\ell|E_j^*E_{j'}|\psi_{\ell'}\rangle = \alpha_{j,j'}\delta_{\ell,\ell'} \ ,$$

pour $(\alpha_{j,j'})_{j,j'}$ un opérateur hermitien.

Pour $I \subseteq \{1,\ldots,n\}$ avec $\#I \leq 2t$ quelconque, considérons l'ensemble Q_I des opérateurs de Pauli qui agissent non-trivialement seulement aux positions dans I. Soit $\{E'_i\}_i := \{\sqrt{\gamma'}W\}_{W\in Q_I}$ la décomposition de Kraus de l'opération quantique $\mathfrak{E}$ qui applique un opérateur de Pauli dans Q_I aléatoirement et uniformément. Nous voulons montrer que C corrige les erreurs produites par $\mathfrak{E}$. Notez que E'_i et $E'_{i'}$ satisfont $W_{i,i'} := E_i^*E_{i'}/\gamma' \in Q_I$. Puisque les opérateurs de Pauli qui agissent sur les positions dans I forment un groupe sous la multiplication, on peut toujours écrire $E_i^*E_{i'}$ comme le produit de deux opérateurs de Pauli agissant chacun sur au plus t positions dans I. Définissons $I_0 \subset I$ qui contient les $\min(\#I, t)$ premières positions de I et $I_1 \subset I$ qui contient les $\max(0, \#I - t)$ positions de I suivantes (i.e. $I_0 \cap I_1 = 0$, $\#I_0 \leq t$ et $\#I_1 \leq t$). Nous pouvons toujours trouver $F_{i,i'} \in Q_{I_0}$ et $F'_{i,i'} \in Q_{I_1}$ tels que $W_{i,i'} = F^*_{i,i'}F'_{i,i'}$. Puisque C sorrige $\{E_i\}_i$ et $\sqrt{\gamma'}F_{i,i'}, \sqrt{\gamma'}F'_{i,i'} \in \{E_i\}_i$, nous avons

$$\begin{aligned}\langle\psi_\ell|E_i'^*E'_{i'}|\psi_{\ell'}\rangle &= \gamma'\langle\psi_\ell|W_{i,i'}|\psi_{\ell'}\rangle \\ &= \langle\psi_\ell|\sqrt{\gamma'}F^*_{i,i'}\sqrt{\gamma'}F'_{i,i'}|\psi_{\ell'}\rangle \\ &= \alpha'_{i,i'}\delta_{\ell,\ell'} \ ,\end{aligned}$$

et C corrige donc pour $2t$ positions connues affectées par des opérateurs de Pauli. Puisque $\{E'_i\}_i$ forme une base pour les opérateurs qui agissent sur les positions dans I, nous concluons par le théorème 42 que C corrige n'importe quelles $2t$ erreurs aux positions I connues. □

Exercice 243. *Montrez que si P est un opérateur de Pauli agissant sur $d-1$ positions alors il peut être écrit comme $P = QR$ où Q agit seulement sur seulement t positions et R agit sur seulement $d-1-t < t$ positions.*

Un *canal quantique à effacement* est un canal quantique qui remplace avec une certaine probabilité un qubit en transit par un état distinct. Par exemple, les qubits dans l'état $|\psi\rangle = \alpha|0000\rangle + \beta|1111\rangle$ après avoir transité sur un canal à effacement peuvent être reçus comme $|\tilde{\psi}\rangle = \gamma_0(\alpha|00*0\rangle + \beta|11*1\rangle) + \gamma_1(\alpha|*00*\rangle + \beta|*11*\rangle)$ où $|*\rangle$ est l'etat correspondant à un symbole d'effacement.

Exercice 244. *Montrez que si C est un qcode qui corrige $d-1$ erreurs arbitraires à des positions connues alors C corrige $d-1$ effacements.*

5.15 Deux bornes sur les codes quantiques

Dans cette section, nous verrons deux bornes simples sur l'existence de codes correcteurs quantiques. Nous verrons la borne du singleton et la borne de Hamming quantique. Ces bornes sont des bornes supérieures, des bornes que les codes quantique doivent satisfaire pour exister. La borne de Hamming est bien adaptée au cas asymptotique, comme sa version classique. La borne du singleton permet, quant à elle, de déterminer l'optimalité d'un code quantique dans certaines conditions. Comme par exemple le code avec le meilleur taux de transmission pour corriger une seule erreur arbitraire.

La première borne est celle de Knill et Laflamme que l'on nomme borne du singleton. Cette borne s'applique à tous les codes, pas seulement à ceux sur un alphabet binaire.

Théorème 43 (borne du singleton)**.** *Si un* $[[n,k,d]]$*–qcode existe alors il doit satisfaire*

$$n-k \geq 2(d-1) \ .$$

Exercice 245. *Prouvez la borne du singleton pour les* $[[n,1,d]]$*–qcodes en transformant un qcode qui est meilleur que la borne du singleton en une machine à cloner les états quantiques, ce qui est impossible.*

La borne du singleton est obtenue en montrant que si un code était tel que $n-k < 2(d-1)$ alors il pourrait être utilisé pour cloner un qubit arbitraire, ce qui n'est pas possible. L'idée est qu'un code quantique qui corrige t erreurs peut être utilisé pour corriger $2t$ erreurs si les positions des erreurs sont connues. Il s'agit d'une version plus sophistiquée de la *borne du non-clonage* qui établit que $n \geq 2(d-1)$. Le prochain corollaire nous indique que pour corriger une erreur arbitraire sur un seul qubit encodé, il est nécessaire d'avoir des *mots de code* de longueur au moins 5.

Corollaire 6. *Un* $[[n,1,3]]$*–qcode doit satisfaire* $n \geq 5$*.*

Il est toujours possible que le code de Shor ait le meilleur taux de transmission pour la correction d'une erreur sur un qubit encodé. Nous verrons que ce n'est pas le cas, nous pouvons faire mieux.

La borne de Hamming quantique est une simple adaptation de la borne de Hamming classique, que nous avons également appelée borne de paquetage de sphères à la sect. 5.6. Cette borne s'applique à tous les codes, peu importe leur alphabet.

Théorème 44 (borne de Hamming quantique)**.** *Si un* $[[n,k,d]]$*–qcode existe alors il doit satisfaire*

$$\left(\sum_{j=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{j} 3^j \right) 2^k \leq 2^n \ .$$

La borne de Hamming quantique peut être exprimée dans sa version asymptotique. Elle résulte en la condition suivante :

Théorème 45 (borne de Hamming quantique asymptotique). *Si un* $[[n,k,2pn]]$*–qcode existe pour* $p \in [0,\frac{1}{2}]$ *alors il doit satisfaire, pour n suffisamment grand,*

$$\frac{k}{n} \leq 1 - \boldsymbol{H}_2(p,1-p) - p\lg(3) \ .$$

La figure 5.2 montre les bornes de Hamming classique et quantique. La borne quantique montre qu'il est impossible de trouver un $[n,k,d]$–qcode avec $\frac{k}{n} > 0$ et $\frac{d}{n} \gtrapprox 0.38$. Autrement dit, aucune code quantique ne peut corriger une taux d'erreur $\frac{t}{n} = \frac{1}{n}\lfloor\frac{d-1}{2}\rfloor \gtrapprox 0.19$ tandis que la borne de Hamming classique n'interdit pas de trouver un code pour corriger un taux d'erreur $\frac{t}{n} < \frac{1}{2}$. La borne du singleton interdit l'existence d'un code quantique à taux de transmission non nul lorsque $\frac{d}{n} \geq \frac{1}{2}$ ce qui donne $\frac{t}{n} \geq \frac{1}{4}$.

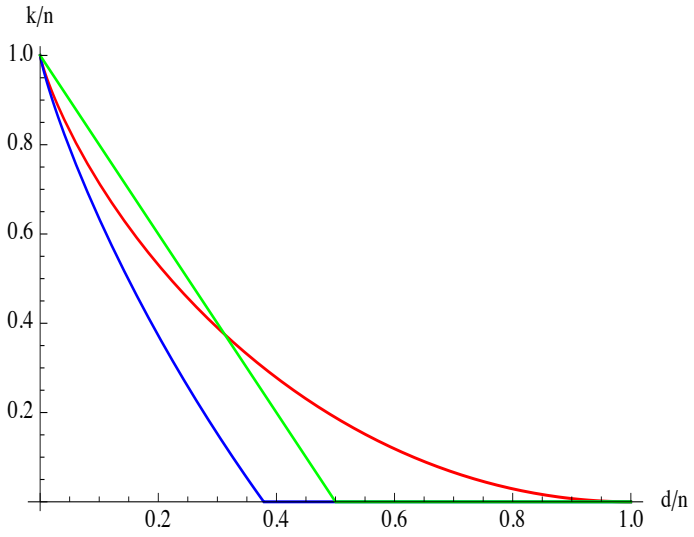

FIGURE 5.2 – Les bornes de Hamming classique(rouge) et quantique(bleu) ainsi que la borne du singleton quantique(vert), des limites sur l'existence de codes.

5.16 Lectures supplémentaires

Le traitement que nous avons fait des codes correcteurs classiques peut être approfondi en consultant le livre de MacWilliams et Sloane[MS78]. Cet ouvrage donne des constructions standards pour des codes correcteurs classiques. Les bornes sur les codes classiques que nous avons vu y sont prouvées formellement. La preuve de la borne de Gilbert-Varshamov pour les codes faiblement duaux est donnée dans l'article de Calderbank et Shor[CS96]. Notre traitement des codes quantiques est semblable à celui du livre de Nielsen et Chuang[NC00]. La preuve du théorème 36 que nous n'avons pas vue y est présentée. Le cours (accessible sur vidéo) que D. Gottesman a donné à IQC (Waterloo) est très bon pour en savoir plus sur les

codes correcteurs quantiques et la tolérance aux fautes de l'ordinateur quantique[Got12]. Un cours à Caltech donné par John Preskill est également très bon pour aller plus loin que nous l'avons fait ici[Pre14].